

Projet Access<>MI

Comment réaliser une liaison MapInfo <> Access

Ludovic Gnemmi

l_gnemmi@yahoo.fr

Octobre 2002

Ce projet vise à familiariser les utilisateurs à la création de liens entre MapInfo et Access et requiert la possession de MapInfo, MapBasic et MSAccess.

Toutes les instructions pour préparer les éléments programmés et les codes nécessaires sont regroupés dans ce document ainsi que les instructions pour faire marcher l'exemple fourni.

Les diverses parties de code sont aussi disponibles sous forme de fichiers indépendants regroupés dans le fichier archive MapInfo_Access.zip avec les fichiers des données utilisées dans l'exemple.

Les applications MapBasic ont été compilées avec la version 5.5 et pour l'exemple fourni. La banque Access « bdbati » (d'une version pré-2000) correspond aussi à l'exemple. La table Mapinfo « bati » lui correspond.

Table des matières

Objectif de cette liaison Access/Mapinfo

Marche à suivre pour préparer la liaison

- mitoaccess.mb
- accesstomi.mb
- accesstomi.txt
- module_ref_mapinfo.txt
- module_info_systeme.txt

Exemple de mise en œuvre

Objectif de cette liaison Access/Mapinfo

1-Visualiser à partir d'un ou plusieurs objets sélectionnés dans MI, le formulaire de(s) l'enregistrement(s) correspondant dans Access grâce à un ID commun, à partir d'un bouton dans MI provoquant l'ouverture du formulaire.

2-visualiser à partir d'un enregistrement en cours dans un formulaire Access l'objet correspondant dans une couche MI grâce à un ID commun

Marche à suivre pour préparer la liaison

(Les divers fichiers indiqués en gras ici sont dans l'archive MapInfo_Access.zip)

Note : pour faciliter l'expérimentation, il est recommandé de créer le répertoire « c:\essai_liaison », d'y placer tous les fichiers de l'archive zip et tous les résultats de compilation.

1-Ouvrir le fichier **mitoaccess.mb** afin de modifier le nom de la table MI, le nom du champ contenant l'ID commun ainsi que le chemin complet de la base Access. Compiler.

2-Ouvrir le fichier **accesstomi.mb** afin de modifier le nom de la table MI, celui du champ comportant l'ID commun ainsi que son type. Compiler.

3-Dans le formulaire Access, ajouter un bouton et insérer le code contenu dans le fichier **accesstomi.txt** dans un événement sur click (faire les modifications nécessaires (nom de table...))

4-Ajouter un nouveau module ref_mapinfo à la base Access et y ajouter le code contenu dans le fichier **module_ref_mapinfo.txt** (faire les modifications nécessaires (nom de table...))

5-Ajouter un nouveau module info_systeme à la base Access et y ajouter le code contenu dans le fichier **module_info_systeme.txt**

6-Ajouter une macro "ref_mapinfo" dans Access avec l'action "Exécuter code" et le nom de fonction "OpenTextFile()"

7-Exécuter Mitoaccess.mbx depuis MI... (voir Exemple de mise en œuvre)

mitoaccess.mb

```
'Programme destiné à connecter les objets d'une couche Mapinfo à
'un formulaire dans MSAccess grâce a un ID commun
'par le biais d'un bouton dans Mapinfo provoquant l'ouverture du
'formulaire correspondant aux objets sélectionnés
'Avant de compiler: Indiquer le nom de la table pour laquelle
'vous souhaitez réaliser la liaison (ligne 10)
'  le nom du champ qui comporte l'identifiant commun (ligne 11)
'  le chemin complet de la base Access à laquelle on veut se
'  connecter(ligne12)

Include "Mapbasic.def"
Include "Menu.def"

Define TabNam "MaTable"           'Ici le nom de La table
Define id_champ "MomChamp"        'Ici le nom du champ contenant
                                  'l'identifiant commun
Define s_filenamebase "C:\...\MaBase.mdb" 'Ici le chemin de la base access

declare sub main
declare sub proc_controle
declare sub proc_connect
declare sub proc_selec

    'La fonction tabl_ouv permet de savoir si une table est ouverte

declare function table_ouv (byval nom_table as string) as logical

    'Les deux fonctions ci-dessous utilisées pour l'ouverture de MSAccess

Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA"
    (ByVal hwnd As integer,
    lpOperation As String,
    lpFile As String,
    lpParameters As String,
    lpDirectory As String,
    ByVal nShowCmd As integer) As integer

Declare Function MiShellExecute
    (ByVal hwnd As integer,
    ByVal lpOperation As String,
    ByVal lpFile As String,
    ByVal lpParameters As String,
    ByVal lpDirectory As String,
    ByVal nShowCmd As integer) As integer

    'La fct ci-dessous est destine a ralentir Mapbasic pendant l'ouverture
    'd'Access lorsque celui-ci est encore fermée

Declare sub Sleep lib "kernel32" Alias "Sleep"(Byval dwMillisecond as integer)

    'La fonction GetTempPath permet d'obtenir le répertoire de rangement
    'des fichiers temp de l'ordi

Declare Function GetTempPath lib "kernel32" Alias "GetTempPathA"
    (ByVal nSize As Integer, IpBuffer As String) as Integer

Define SHOWMAXIMIZED 3
Define SHOWMINIMIZED 2
```

```

Define SHOWNORMAL 1
.....

Function table_ouv (byval nom_table as string) as logical
dim i as smallint
  for i=1 to numtables()
    If TableInfo(i,TAB_INFO_NAME)=nom_table then
      table_ouv=true
      exit function
    End If
  Next
end function
.....

Function MiShellExecute (ByVal hwnd As integer,
ByVal lpOperation As String,
ByVal lpFile As String,
ByVal lpParameters As String,
ByVal lpDirectory As String,
ByVal nShowCmd As integer) As integer

MiShellexecute = ShellExecute (hwnd, lpOperation, lpFile, lpParameters,
lpDirectory, nShowCmd)
end function

.....

Sub main
Alter ButtonPad ID 4
  add Pushbutton
    id 100
    icon 192
    helpmsg "Voir la fiche descriptive de l'objet sélectionné " +
      "dans access\nFormulaire Access"
    calling proc_controle
end Sub
.....

      'Et si ce n'est pas le cas:  alors ouverture de la table
Sub proc_controle
dim s_filename as string
If table_ouv(TabNam)=1 then
  call proc_selec
Else
  s_filename = "Open Table "+Chr$(34)+FileOpenDlg("C:\",TabNam,"TAB",
    "Veuillez situer la table "+TabNam)+Chr$(34)+" Interactive"
  if s_filename="Open table "+Chr$(34)+Chr$(34)+" Interactive" then
    exit sub
  End If
  Run Command s_filename
  If table_ouv(TabNam)<>1 then
    note "La table "+TabNam+" n'est pas ouverte"
    Exit sub
  End If
  If NumWindows()=0 then
    Map From TabNam
  Else
    Add Map Auto Layer AFP
  End If
  Call proc_selec
End If

End Sub

.....

      'Procédure d'écriture de l'ID de l'objet sélectionné dans un fichier txt
Sub proc_selec

```

'Ci-dessous, lignes de code utilisées pour localisation du temp dir

```
Dim x as integer
Dim TempPath as string
Dim IpBuffer as string
Dim buffSize as Integer
Dim i, j as integer

BuffSize=1024
IpBuffer=string$(1024,chr$(32))
x=GetTempPath(BuffSize,IpBuffer)

tempPath=IpBuffer
TempPath=TempPath+"transfert.txt"

j=Columninfo(tabnam, id_champ,COL_INFO_NUM)

If SelectionInfo(SEL_INFO_NROWS)<>0 and
selectioninfo(SEL_INFO_TABLENAME)=TabNam then
    Open File TempPath For OUTPUT As #1
    Fetch First From selection
    Do While Not EOT(selection)
    Print #1, Selection.col(j)
    Fetch Next From selection
    Loop
    Close File #1
Else
    note "Veuillez sélectionner un ou plusieurs objets et " +
        "cliquer à nouveau sur le bouton"
    Exit sub
End If
    call proc_connect
End sub
.....
'Procédure d'ouverture de la base access (si elle n'est pas ouverte)
'et d'exécution de la macro ref_mapinfo

Sub proc_connect

Dim canal, ii as integer

Onerror goto docnonouvert

canal = ddeinitiate("MSAccess", s_filenamebase)
reprise:
ddeexecute canal , "ref_mapinfo"
ddeterminate canal
Exit sub

docnonouvert:
If err()=697 then
    ii = MiShellexecute (0, "open", s_filenamebase, "", "", SHOWNORMAL)
    'pour faire attendre Mapbasic 2s pendant l'ouverture de la base Access
    call sleep(2000)
    canal = ddeinitiate("MSAccess", s_filenamebase)
    Resume reprise
end If
End sub
```

accesstomi.mb

```
'Programme destiné à visualiser a partir d'un enregistrement en cours
'dans un formulaire MSAccess l'objet correspondant dans une couche MI
'grâce à un ID commun
'Avant de compiler:Indiquer le nom de la table MI pour laquelle on veut
'réaliser la liaison (ligne 10)
' le nom du Champ dans MI qui comporte l'ID commun (ligne 11)
' ainsi que son type en modifiant le type de la variable r (ligne 12)

Include "MapBasic.def"
Include "Menu.def"

Define NamTab "Ma table"          'Ici le nom de la table MI
Define nom_champ "Mon champ"      'Ici le nom du champ contenant l'ID commun
Dim r as integer                 'Ici le type du champ utilise pour la liaison
                                '(Integer ou string)

Declare sub main
declare sub RemoteMsgHandler
declare sub proc_centrer

'La fonction tabl_ouv permet de savoir si une table est ouverte

declare function table_ouv (byval nom_tab as string) as logical

.....
sub main

end sub
.....
Function table_ouv (byval nom_tab as string) as logical
dim i as smallint
for i=1 to numtables()
    If TableInfo(i,TAB_INFO_NAME)=nom_tab then
        table_ouv=true
        exit function
    End If
Next
end function
.....
'Procédure récupérant l'identifiant envoyé par Access.
Sub RemoteMsgHandler
Set window WIN_MAPINFO Front
dim s_filename as string
'recupération dans r du message envoyé par Access
r=CommandInfo(CMD_INFO_MSG)
'si la table est déjà ouverte, appel de la proc_center
'sinon ouverture de la couche
If table_ouv(NamTab)=1 then
    call proc_centrer
Else
    s_filename = "Open Table "+Chr$(34)+FileOpenDlg("C:\",NamTab,"TAB",
        "Veuillez situer la table "+NamTab+Chr$(34)+" Interactive"
    if s_filename="Open table "+Chr$(34)+Chr$(34)+" Interactive" then
        exit sub
    End If
    Run Command s_filename
    If table_ouv(NamTab)<>1 then
        note "Action annulée"
    Exit sub
End If
```

```

        End If
        If NumWindows()=0 then
            Map From NamTab
        Else
            Add Map Auto Layer AFP
        End If
        Call proc_centrer
    End If

end sub

'zoom sur l'objet sélectionné
Sub proc_centrer
Dim j as string

j="Select * from "+NamTab+" where "+NamTab+"."+nom_champ+"="

    Run command j+r
    If SelectionInfo(SEL_INFO_NROWS)=1 then
        Set Map
        Window frontwindow()
        Center (centroidx(selection.obj),centroidy(selection.obj))
    Else
        note "Aucune selection correspondante"

    End If
    statusbar hide
    statusbar show

end sub

```


accesstomi.txt

```
'Code à ajouter dans un nouveau bouton à placer dans le formulaire
'Access pour l'événement click
    'Attention : Indiquer le chemin exact de l'appli accesstomi.mbx
    'Et le nom du champ dans le form Access

Private Sub activate_mi_Click()

Dim canal, canal2 As Long
Dim Pathmbx, str As String
Dim i As variant

    'Indiquer le nom du champ du form Access comportant l'ID

i = [NOM_CHAMP]
str = Chr$(34)

    'Changer le chemin exact de l'appli accesstomi.mbx

Pathmbx = "C:\Program Files\MapInfo\MapBasic\...\accesstomi.mbx"

On Error Resume Next

canal = DDEInitiate("Mapinfo", "system")
If Err.Number <> 0 Then
    Err = 0
    Shell "C:\Program Files\MapInfo\Professional\Mapinfow.exe"
    canal = DDEInitiate("Mapinfo", "system")
End If

DDEExecute canal, "Run application " & str & Pathmbx & str

canal2 = DDEInitiate("Mapinfo", Pathmbx)
DDEExecute canal2, i

DDETerminate canal
DDETerminate canal2

End Sub
```

module_ref_mapinfo.txt

'Code à ajouter dans un nouveau module "ref_mapinfo" de la base Access
'Attention : Indiquer le nom du formulaire Access et
'le nom du champ comportant l'ID commun

Option Compare Database

Option Explicit

```
Public Function OpenTextFile()  
Const ForReading = 1, ForWriting = 2, ForAppending = 3  
Dim fs, a, refup, vartab()  
Dim intstrOpenArg As Integer  
Dim strTempFile, strOpenArg, strOpenArgb, strOpenArgtxt As String  
Dim i As Integer  
  
i = 1  
    'Référence à la fonction du module ref_systeme qui donne le  
    'chemin du reper temp  
strTempFile = fReturnTempDir()  
strTempFile = strTempFile + "Transfert.txt"  
Set fs = CreateObject("Scripting.FileSystemObject")  
Set a = fs.OpenTextFile(strTempFile, ForReading, False)  
    'Indiquer le nom du champ Access correspondant à l'ID commun  
    '(remplacer ID_BAT)  
strOpenArg = "[ID_BAT]="  
strOpenArgb = strOpenArg  
Do While a.AtEndOfStream <> True  
    ReDim vartab(i)  
    vartab(i) = a.readline  
    'Indiquer le nom du champ Access correspondant à l'ID commun  
    '(remplacer ID_BAT)  
    strOpenArg = strOpenArg + vartab(i) + " Or [ID_BAT]="  
    i = i + 1  
    ReDim vartab(i)  
Loop  
intstrOpenArg = Len(strOpenArgb) + 4  
strOpenArgtxt = Mid(strOpenArg, 1, (Len(strOpenArg)) - intstrOpenArg)  
    'Indiquer le nom du formulaire Access (ici batiments)  
DoCmd.OpenForm "batiments", , , strOpenArgtxt  
End Function
```

module_info_systeme.txt

```
'Code à ajouter dans un nouveau module "info_systeme" de la base Access
'Option Compare Database
'Fonction destinee a connaitre le repertoire temp.

'***** Code Start *****
Private Const MAX_PATH As Integer = 255

Private Declare Function apiGetTempDir Lib "kernel32" Alias "GetTempPathA"
(ByVal nBufferLength As Long,
ByVal lpBuffer As String) As Long

'*****
Function fReturnTempDir()
'Retourne le répertoire temporaire
Dim strTempDir As String
Dim lngx As Long
strTempDir = String$(MAX_PATH, 0)
lngx = apiGetTempDir(MAX_PATH, strTempDir)
    If lngx <> 0 Then
        fReturnTempDir = Left$(strTempDir, lngx)
    Else
        fReturnTempDir = ""
    End If
End Function

'***** Code End*****
```

Exemple de mise en œuvre

1 - Si le répertoire "C:\essai_liaison" n'existe pas, il faut le créer.

2 – Ce répertoire doit comprendre « AccessToMi.MBX », « MiToAccess.MBX », « BDBati.MBD », « Bati.TAB ... »

3 - Ouvrir MI s'il ne l'est pas déjà

4 - Dans MI,

a/ lancer l'application MiToAccess.mbx. Une icône (paire de lunettes) est ajoutée à la barre d'icônes OUTILS.

b/ ouvrir la table bati.tab

5 - De l'Explorateur Windows, lancer directement "bdbati.mdb". Access va s'ouvrir sur la feuille d'enregistrement contenant entre autres choses le bouton "Visualiser dans MI"

6 - Disposer les 2 fenêtres de façon à en voir le contenu complet et commencer à "jouer"

a/ dans MI, sélectionner un ou plusieurs objets, puis activer l'icône "paire de lunettes". Access affiche sur la première ligne les données correspondantes au(x) bâtiment(s) choisi(s).

b/ dans Access, sélectionner un des cas et activer le bouton "Visualiser dans MI". Le bâtiment correspondant s'affiche comme sélectionné au centre de la fenêtre carte.