



Extrait du Environnement iSeries

<http://xdocs400.com/spip.php?article241>

# Utiliser les composants OLE fournit par Iseries Access

- Les articles -



Date de mise en ligne : vendredi 29 avril 2005

---

**Environnement iSeries**

---

### Exemple utilisant les controles fournis avec client ACCESS et permettant de faire "dialoguer" vos applications ISERIES avec EXCEL (VBA).

Le transfert du contenu de la feuille Excel se fait en 2 étapes :

- Transfert du contenu d'une feuille Excel vers une DTAQ
- Appel (à partir de la feuille Excel) d'un programme ISERIES chargé de copier le contenu de la DTAQ vers un fichier physique.

#### ➤ Fichier Physique FICXLS

```
CREATE TABLE BIBXLS/FICXLS (CODE DEC ( 6, 0) NOT NULL WITH DEFAULT, LIBEL CHAR (35 ) NOT NULL WITH DEFAULT, GENCOD DEC (13 , 0) NOT NULL WITH DEFAULT)
```

#### ➤ DTAQ RECEPTXLS

```
CRTDTAQ DTAQ(BIBXLS/RECEPTXLS) MAXLEN(1000) TEXT('DtaQ réception XLS')
```

#### ➤ Feuille Excel Envoi

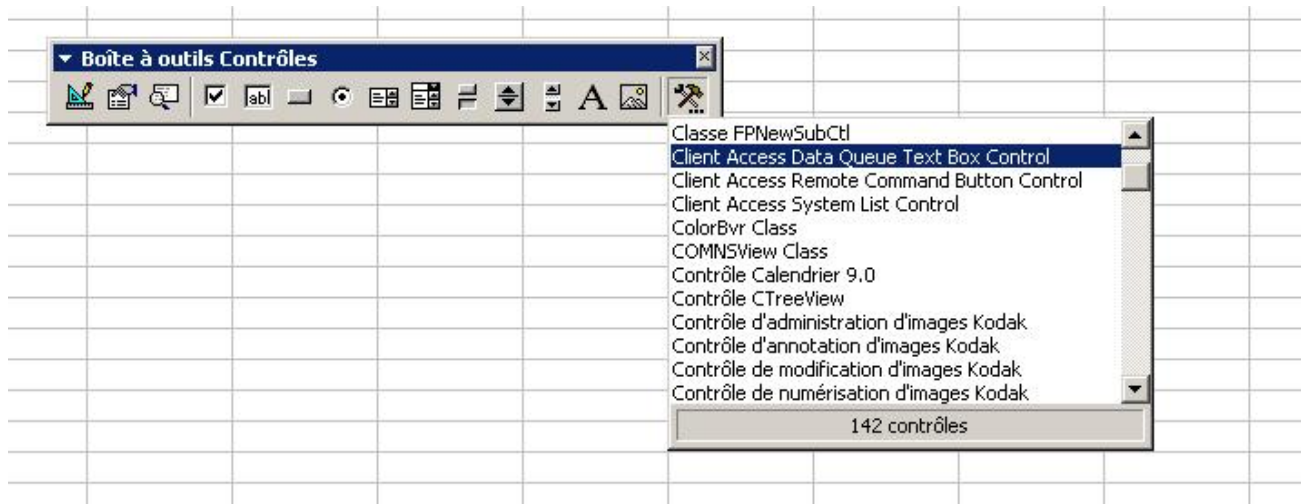
Code	Libellé	Gencod
936324	CADRE A PETITS BOIS MS52 DE 1308 3	1234569999999
936326	CADRE A PETITS BOIS MS52 DE 1312 3	1234566666666
936332	CADRE A PETITS BOIS MS52 DE 1408 3	1234567474747
936334	CADRE A PETITS BOIS MS52 DE 1412 3	1234567987979
936336	CADRE A PETITS BOIS MS52 DE 1415 3	1234561245475
936341	CADRE A PETITS BOIS MS52 DE 1610 3	1234561245412
936342	CADRE A PETITS BOIS MS52 DE 2008 3	1234568978941
936348	CADRE A PETITS BOIS MS52 DE 2024 3	1234568778787
936352	CADRE A PETITS BOIS MS40 DE 760 3	9876541354648
936356	CADRE A PETITS BOIS MS40 DE 780 3	6543217899651
936370	CADRE A PETITS BOIS MS40 DE 1150 3	6543211235478
936375	CADRE A PETITS BOIS MS40 DE 1175 3	5465468946119
936384	CADRE A PETITS BOIS MS40 DE 1280 3	1000000000009
936389	CADRE A PETITS BOIS MS40 DE 1365 3	1237891212121
936394	CADRE A PETITS BOIS MS40 DE 1450 3	1237891212122

#### Pré-Requis

- Pour cet exemple j'ai utilisé Excel 2000 (sous windows) et la version OS400 V5R2
- Pour disposer des contrôles de transmission de IBM iSeries Access for Windows (avec VB ou VBA), vous devez installer la boîte à outil de programmation fournit avec la version client de IBM iSeries Access for Windows. Vous choisirez une installation personnalisée du produit, en prenant soin de sélectionner l'option « boîte à outils de programmation ».

#### Utilisation des controles avec Excel

- Afficher la barre d'outil VB (menu outils-personnaliser)
- Passez en mode création
- Les controles qui nous interesse sont disponibles dans le menu autres controles.



### Codage VBA

2 fonctions gèrent l'essentiel du traitement.

- Alim400 qui permet de scanner notre feuille excel et de transférer les données vers la DTAQ en utilisant le contrôle CwbDataQueueTextBox.
- exec400 se charge d'appeler le programme AS400 chargé de copier le contenu de la DTAQ vers un fichier physique. Cette fonction utilise un contrôle CwbRemoteCommandButton. Le choix de l'AS400 se fera par sélection dans une liste par le contrôle CwbSystemListBox .

Vous pouvez placer ces fonctions dans le code de la feuille ou dans un module séparé.

```
Private Sub CommandButton1_Click()  
    If Not (Alim400) Then Exit Sub  
    If (exec400) Then MsgBox "transfert OK"  
End Sub
```

### CwbSystemListBox : Boite à liste des systèmes :

Ce contrôle personnalisé a l'apparence d'une list-box permet d'afficher la liste des serveurs, dans laquelle un utilisateur peut effectuer son choix. La liste des serveurs affichée peut faire apparaître tous les serveurs configurés (Op Navigator), tous les serveurs connectés ou tous les serveurs qui ne sont pas connectés. Cette liste peut être régénérée périodiquement, de manière à maintenir à jour les informations affichées.

- Code VBA permettant de récupérer le serveur sélectionné.

```
Dim rmtAs As String  
rmtAs = Envoi.cwbSystemListBox1.GetItemText(cwbSystemListBox1.CurrentSel)
```

### CwbDataQueueTextBox : Zone de texte de file d'attente de donnée :

Ce contrôle a l'apparence d'une zone d'édition et est utilisé pour indiquer des données texte à transférer vers une DTAQ. Il permet également de lire les données d'une DTAQ et de les afficher comme des données textes. Les données textes peuvent également être converties automatiquement entre ASCII et EBCDIC .

➤

Job associé (visible par NETSTAT) 10.132.21.198 1331 as-dtaq 000:00:20 Established

➤ Les propriétés, méthodes et événements ci-après sont associés à ce contrôle. Envoi de données vers l'AS400 Pour l'envoi de donnée la propriété

&mdash; TransferType = 0 (Envoi)

&mdash; ConvertData = true (ASCII-EBCDIC)

&mdash; LibraryName = BIBXLS

&mdash; QueueName = RECEPTXLS

&mdash; SystemName = 192.28.10.1 (vous pouvez utiliser le ctl CwbSystemListBox )

&mdash; MaxLength = 1000

➤ Routine VBA Alim400 :

```
Function Alim400() As Boolean
' Cette fonction renvoie true si le transfert c'est bien passée
Dim retVal, x_lig, y_col, i As Long
Dim ligTxt As String
Alim400 = False
x_lig = 2 'On commence à la 2ème ligne
Do While (Envoi.Cells(x_lig, 1) <> "") 'Tant que la 1ère colonne est renseignée
ligTxt = ""
ligTxt = CStr(Envoi.Cells(x_lig, 1)) & ";" & CStr(Envoi.Cells(x_lig, 2)) _
& ";" & CStr(Envoi.Cells(x_lig, 3))

' Alimentation de la DTAQ

Envoi.cwbDataQueueTextBox1.Text = ligTxt
retVal = Envoi.cwbDataQueueTextBox1.TransferTextData
If retVal <> 0 Then
MsgBox sErr & "erreur détectée ligne " & x_lig
Exit Function
End If
x_lig = x_lig + 1
Loop
'écriture de la dernière ligne de la DTAQ
Envoi.cwbDataQueueTextBox1.Text = "FIN_TRANSF"
retVal = Envoi.cwbDataQueueTextBox1.TransferTextData
If retVal <> 0 Then
MsgBox sErr & "erreur détectée sur la ligne de fin de la DTAQ"
Exit Function
End If
Alim400 = true
End Function
```

### **CwbRemoteCommandButton : Bouton de commande à distance :**

Ce contrôle (à l'apparence d'un bouton de commande) peut être utilisé pour exécuter des commandes sur un serveur. Lorsque l'utilisateur appuie sur ce bouton, la commande spécifiée est envoyée au serveur. Après le lancement de la commande, il est possible d'obtenir des informations sur son exécution. Pour notre exemple ce contrôle sera entièrement piloté par programmation. La propriété visible sera à false et nous invoquerons l'exécution avec la méthode "DoClick".

➤

## Utiliser les composants OLE fournis par Iseries Access

Routine VBA exec400 :

```

Function exec400()
    Dim sErr, sTempErr As String
    Dim i, retval As Long
    exec400 = False
    Envoi.cwbRemoteCommandButton1.CommandString = "CALL PGM(SERGE/RECEPTXLS)"
    Envoi.cwbRemoteCommandButton1.Locked = False
    Envoi.cwbRemoteCommandButton1.Enabled = True
    Envoi.cwbRemoteCommandButton1.DoClick 'Exécute le programme sur l'AS400
    retval = Envoi.cwbRemoteCommandButton1.GetErrorMessageCount 'Récupère le nombre de messages d'erreurs et les
    affiches.
    If retval > 0 Then
        For i = 1 To retval
            Envoi.cwbRemoteCommandButton1.GetErrorMessageText i, sTempErr
            sErr = sErr & sTempErr & vbCrLf
        Next
        MsgBox sErr
    Exit Function
    End If
    exec400 = True
End Function
    
```

### Programme AS400 de lecture de la DTAQ

```

* Serge GOMES Lecture DTAQ RECEPTXLS          * ----- FFICXLS O E K DISK rename(FICXLS:FICZ)
* Data Queue Variables                          D RcvQueueName S 10 Inz('FILEA') D RcvQueueLib S 10 Inz('SERGE') D
RcvMsgSize S 5 0 Inz(%Size(RcvMsg)) D RcvMsg S 1000 D RcvWaitTime S 5 0 Inz(10) * Variables de
travail D Separ C ';' D dep S 5 0 Inz(0) D fin S 5 0 Inz(0)
* ----- C ExSr RcvDtq C ExSr TrfDtq C Dow RcvMsg
<> 'FIN_TRANSF' and RcvMsg <> *blan C ExSr RcvDtq C ExSr TrfDtq C EndDo
C eval *INLR = '1' * ----- C RcvDtq BEGSR C Eval
RcvMsg = *Blank C CALL 'QRCVDTAQ ' 90 C PARM RcvQueueName
C PARM RcvQueueLib C PARM RcvMsgSize C PARM RcvMsg
C PARM RcvWaitTime C ENDSR *
----- * C TrfDtq BEGSR C If RcvMsg <> *BLANK and RcvMsg
<> 'FIN_TRANSF' C Eval dep = 1 C Eval fin = 0 C Eval
fin=%scan(Seper:RcvMsg:dep) - 1 C Eval code = %dec(%subst(RcvMsg:dep:fin):6:0) C Eval dep = fin + 2
C Eval fin=%scan(Seper:RcvMsg:dep) - 1 C Eval libel= %subst(RcvMsg:dep:fin) C Eval dep = fin +
2 C Eval gencod = %dec(%subst(RcvMsg:dep:13):13:0) C Write FICZ C
Endlf C ENDSR
    
```