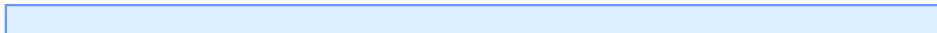


Tutoriel 19 : Accès aux bases de données "ADO" avec VB 6.0

par Gilbert Miralles (gilmir.developpez.com)

Date de publication : Lundi 18 mars 2003

Dernière mise à jour : Lundi 4 février 2008



I - Introduction

II - Le modèle ADO

II-1 - Modèle d'objet "ADO"

II-2 - Environnement des bases de données

III - Téléchargements

Prochain tutoriel :

I - Introduction

La gestion des fichiers a toujours été complexe.

La multiplicité des formats qui sont apparus au gré des versions de Visual Basic ont rendu la tâche des programmeurs de plus en plus difficile.

Visual Basic donne accès aux fichiers à travers le moteur de données Microsoft Jet Engine, l'interface DAO (Data Access Object) et l'ODBC (Open DataBase Connectivity).

Les contrôles de données (DAO) se sont enrichis progressivement, après la version 3.0 d'une zone de liste et d'une boîte combo spécialisée base de données, ainsi que d'une grille de base de données.

Ces contrôles spécialisés sont précédés d'un préfixe : dbList, dbCombo et dbGrid.

Les DAO (objets d'accès aux données) vous permettent d'utiliser un langage de programmation pour extraire et manipuler des données dans des bases de données locales ou distantes.

Ils vous permettent également de gérer des bases de données, ainsi que les objets et la structure de ces dernières.

Les anciennes versions de VB utilisaient les moteurs "Jet" "DAO" et RDO".

Avec l'arrivée de VB6 un nouveau moteur de bases de données est né et a été baptisé : **"ADO" (ActiveX Data Objects)**.

Nous allons étudier sur ces nouvelles technologies pour réaliser dorénavant nos bases de données.

II - Le modèle ADO

Dans les versions précédentes de Visual Basic, il était possible de lier les contrôles entre-eux uniquement sur une feuille.

En revanche, la version 6.0 de Visual Basic permet de lier toute source de données à un utilisateur de données.

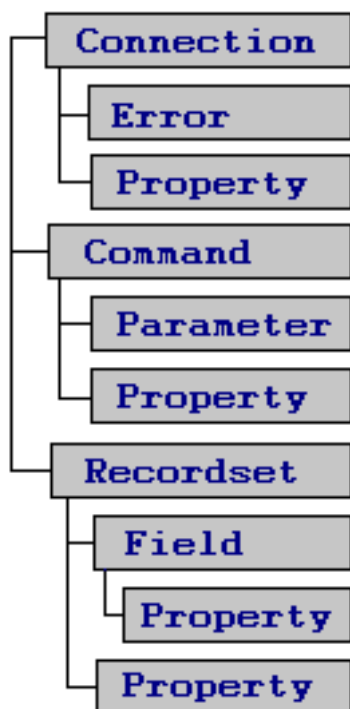
Il est possible de définir la propriété DataSource des contrôles au moment de la création pour établir une liaison dynamique avec les sources de données.

Vous avez la possibilité de créer des classes qui se comporteront comme des sources et des utilisateurs de données et de les lier grâce au nouvel objet BindingsCollection.

Il est également possible de produire des contrôles utilisateur qui sont des sources de données, similaires au contrôle de données ADO.

Vous pouvez ainsi générer des contrôles utilisateur connectés par une liaison complexe, identiques au contrôle DataGrid.

II-1 - Modèle d'objet "ADO"



Le nouveau moteur de BdD de type "ADO" est constitué de sept objets dont les relations structurelles sont décrite sur la figure ci-jointe.

L'accès à une source de données est réalisée par l'intermédiaire d'une nouvelle interface nommée OLE-DB.

Nous disposons avec cet environnement d'une souplesse de construction qui offre des performances plus élaborées que celles offertes par les versions précédentes.

II-2 - Environnement des bases de données


Lorsque vous ajoutez un **Data Environnement** (environnement de base de données) à un projet, Visual Basic crée automatiquement un objet "Connection ADO", dont les propriétés, Nom, Type de base de données, droits d'accès, etc... peuvent être définis de manière similaire à celle d'un contrôle.

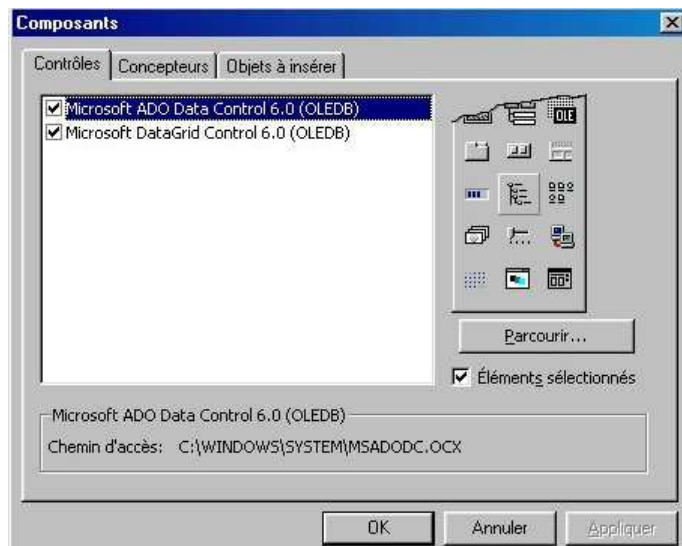
Cette connexion est alors établie pour permettre de transférer des informations de structure à partir de la base de données.

Une des grandes nouveautés de Visual Basic 6.0 est la possibilité d'établir des relations entre les objets "Command" d'une connexion, de manière à constituer et à pouvoir appeler ainsi des dépendances hiérarchiques entre des données avec lesquelles des relations existent.

Nous allons travailler sur un exercice pour vous démontrer qu'il est possible d'accéder aux informations contenues dans une base de données sans écrire la moindre ligne de code.

Nous allons récupérer dans les fichiers stockés dans l'environnement de Visual Basic un fichier intitulé "Biblio.mdb", si vous ne le possédez pas, je vous donne la possibilité de le télécharger sur mon site. [Télécharger le fichier](#)

 *Ce fichier sera stocké dans le répertoire de travail du projet que vous allez réaliser*



Lancez la procédure pour créer un nouveau projet.

Par la procédure décrite précédemment insérez dans la boîte à outils le contrôle MSADODC nécessaire à la réalisation de notre base de données ADO

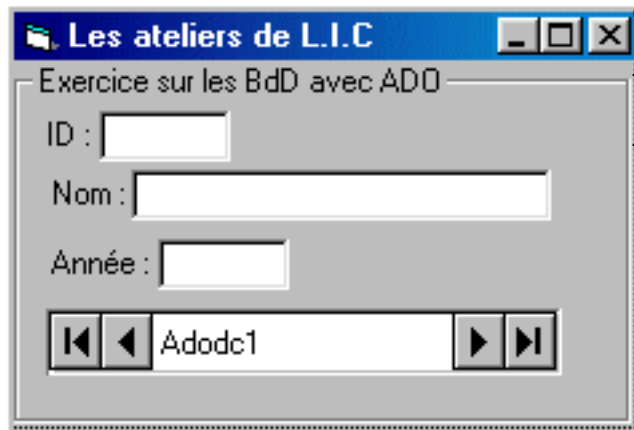
Voici l'image



de l'icône du contrôle "ADODC" que vous devez avoir inséré dans votre boîte à outils.

Réalisation de notre interface graphique devant nous servir d'exemple :

Par la méthode habituelle créez un nouveau projet, insérez dans ce projet le contrôle ADODC comme expliqué précédemment et réalisez l'interface graphique indiqué sur la figure ci-dessous.



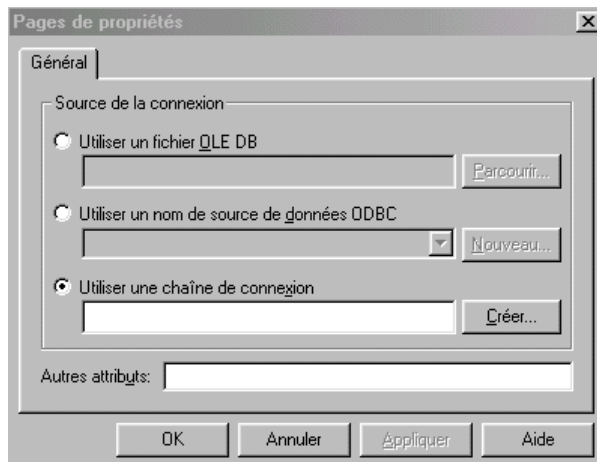
Ce projet comprend :

Trois étiquettes de Label1 à Label3

Trois zones de texte de Text1 à Text3

Un contrôle ADODC intitulé Adodc1

Vue des valeurs de propriétés disponibles dans la feuille "ConnectionString" :

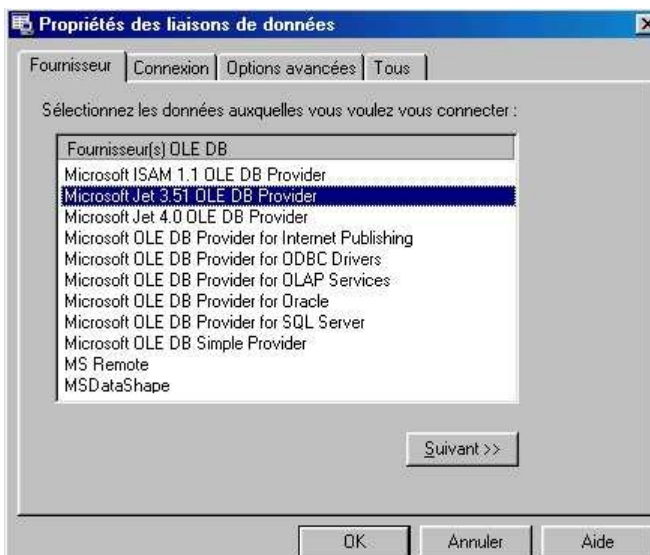


Les informations nécessaires pour réaliser l'interconnexion à la base de données sont stockées dans la propriété "ConnectionString" du contrôle ADODC.

Dans la fenêtre de propriétés de ce contrôle sélectionnez l'option "**Utiliser une chaîne de connexion**" puis cliquez sur le bouton "**Créer...**"

Ce qui vous ouvre la boîte de dialogue ...

Propriétés des liaisons de données :



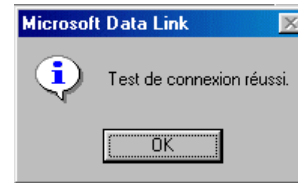
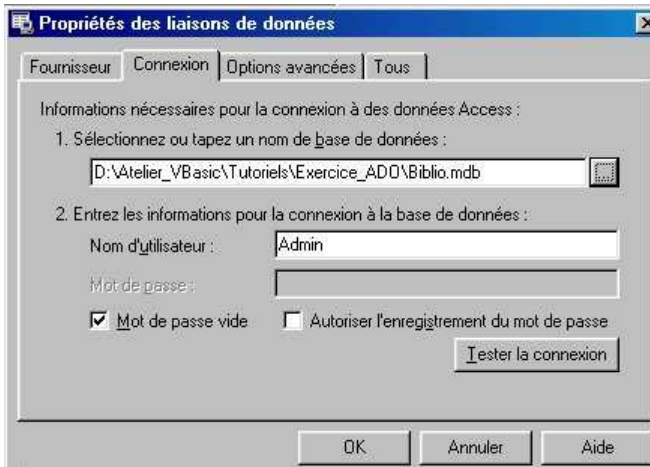
Nous sélectionnons l'étiquette intitulée "Microsoft Jet 3.51 OLE DB Provider" et cliquons ensuite sur le bouton marqué "**Suivant**" pour ouvrir la feuille des propriétés des liaisons de données qui va vous permettre de tester la connexion entre l'application et le fichier de base de données.

Vous pouvez maintenant tester la connexion :

Ce qui vous permet de vérifier que celle-ci peut être établie.

Vu de la boîte de dialogue "Propriétés des liaisons de données" après la sélection du fichier "Biblio.mdb"

Si tout se passe bien voici ce que vous devez obtenir...



Je pense que vous avez compris que vous devez indiquer le chemin de l'endroit où vous avez stocké le projet qui contient aussi le fichier Biblio.mdb, mais j'exagère en vous le rappelant, désolé!

Il ne nous reste plus qu'à renseigner les valeurs des propriétés comme suit :

Propriétés du Contrôle ADO DC :

Name : adoBiblio

Caption : Auteurs

CommandType : 2 - adCmdTable

ConnectionString : Provider=Microsoft Jet... 'Cliquez sur l'étiquette et sélectionnez...

RecordSource : Authors

Propriétés de la zone de texte (Text1)

Name : txtID

DataField : AU_ID **DataSource** : adoBiblio

Propriétés de la zone de texte (Text2)

Name : txtNom

DataField : Author

DataSource : adoBiblio

Propriétés de la zone de texte (Text3)

Name : txtAnnéeN

DataField : Year Born

DataSource : adoBiblio

Les noms des Label n'ont aucune influence sur le programme et je vous laisse le soin de les renommer conformément aux pratiques usuelles dans les pages et exercices divers que je diffuse sur Internet.



Une vue de l'application en fonctionnement

Cette application a été réalisée avec en annexe un fichier nommé "Biblio.mdb".

Ce fichier est trop volumineux pour être téléchargé, mais si vous possédez Visual Basic 6.0, vous l'avez automatiquement dans le répertoire "Program Files/Microsoft Visual studio".

- Faites une recherche avec, "Rechercher :" depuis le lecteur où est installé "Visual Studio".
- Créez à la racine de votre disque dur ("C:\") un répertoire nommé : "Atelier Vbasic\Tutoriels\Exercice_ADO\"
- Insérez dans ce répertoire le fichier "Biblio.mdb"
- Installez le fichier Exo_Ado.zip en le dézipant dans ce même répertoire, et cela doit fonctionner.

III - Téléchargements

Télécharger l'application Exo_Ado.zip (sans le fichier "Biblio.zip")

Télécharger l'application Exo_Ado.zip (avec le fichier "Biblio.zip")

Prochain tutoriel :

 ***Convertir les programmes écrits sous VB3 pour VB6.0***

