

Gestion de photos par formulaire

par [Charles A. \[cafeine\]](#)

Date de publication : 22/10/2004

Dernière mise à jour : 22/10/2004

Gérer des fichiers de photos externes depuis Access par un formulaire.

- 1- Introduction
- 2- Rendu final et utilisation
- 3- Structure des données
- 4- Conception du formulaire
 - a. Génération automatique
 - b. Ajout des contrôles
- 5- Codage du formulaire
 - a. Affichage des photos
 - b. Ajout - Suppression des photos
- 6- Conclusion

1- Introduction

On constate sur le [forum access](#) un grand nombre de questions relatives à la gestion des images.

Il est vrai que MS-Access sait gérer et stocker des fichiers images dans ses tables, mais il n'est pas spécialisé dans ces fonctions.

On pourra ainsi lire très souvent qu'il faut adopter une méthode différente.

Personnellement, je préconise de laisser les fichiers à l'extérieur de la base, en les stockant dans un sous-répertoire dédié.

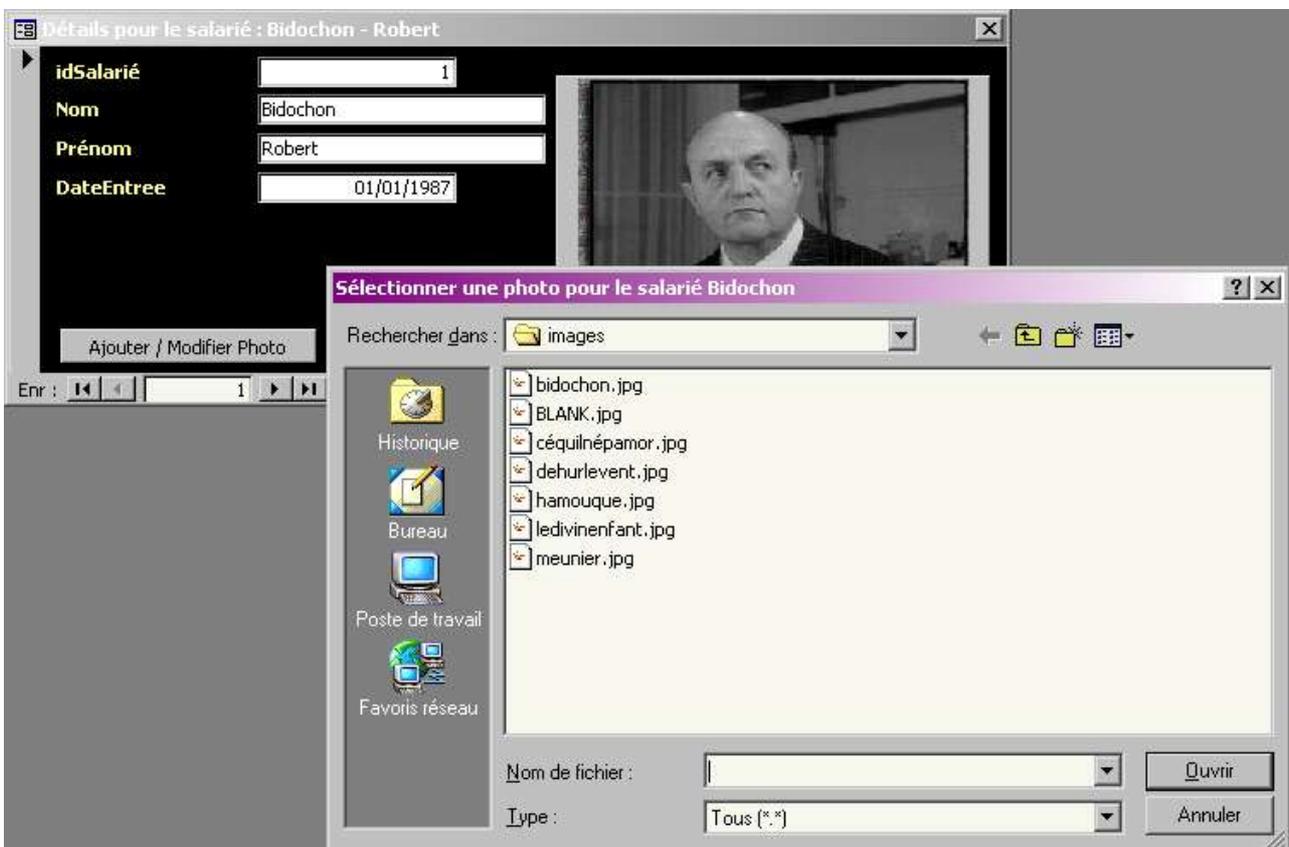
Ce tutoriel se propose de présenter un cas très simple de gestion des images externes à la base.

2- Rendu final et utilisation



Voici le rendu final, du tutoriel, un formulaire qui recense les salariés d'une PME avec l'affichage d'une photo pour chacun d'eux.

On trouvera sur le formulaire, deux boutons de commande pour attribuer / effacer une photo d'un salarié.



En cliquant sur le bouton "Ajouter / Modifier Photo" on pourra parcourir les disques pour pointer sur la photo de l'employé.

3- Structure des données

L'application réduite à son minimum se contentera d'une table : tblSalaries : Table des salariés

tblSalaries : Table		
	Nom du champ	Type de données
	idSalarié	NuméroAuto
	Nom	Texte
	Prénom	Texte
	DateEntree	Date/Heure
	Photo	Texte

Général		Liste de choix
Taille du champ	255	
Format		
Masque de saisie		
Légende		
Valeur par défaut		
Valide si		
Message si erreur		
Null interdit	Non	
Chaîne vide autorisée	Oui	
Indexé	Non	
Compression unicode	Oui	

On a défini le champ **Photo** de la table tblSalaries en type Texte.

Au lieu de stocker physiquement la photo dans la base/table, on va se contenter de stocker le chemin de son emplacement physique.

Cette information sera stockée dans le champ **Photo**, sous forme de chaîne de caractères.

tblSalaries : Table				
	idSalarié	Nom	Prénom	DateEntree
▶	1	Bidochon	Robert	01/01/1987
	2	Meunier	Tudor	14/07/1992
	3	Hamouque	Nadine	15/08/1996
	4	Céquilnépamort	Sylvie	11/11/2000
	5	Ledivinenfant	René	25/12/2003
	7	Dehurlevent	Léo	31/12/1999
*	(NuméroAuto)			

Voici les quelques données contenues dans la table **tblSalaries**.

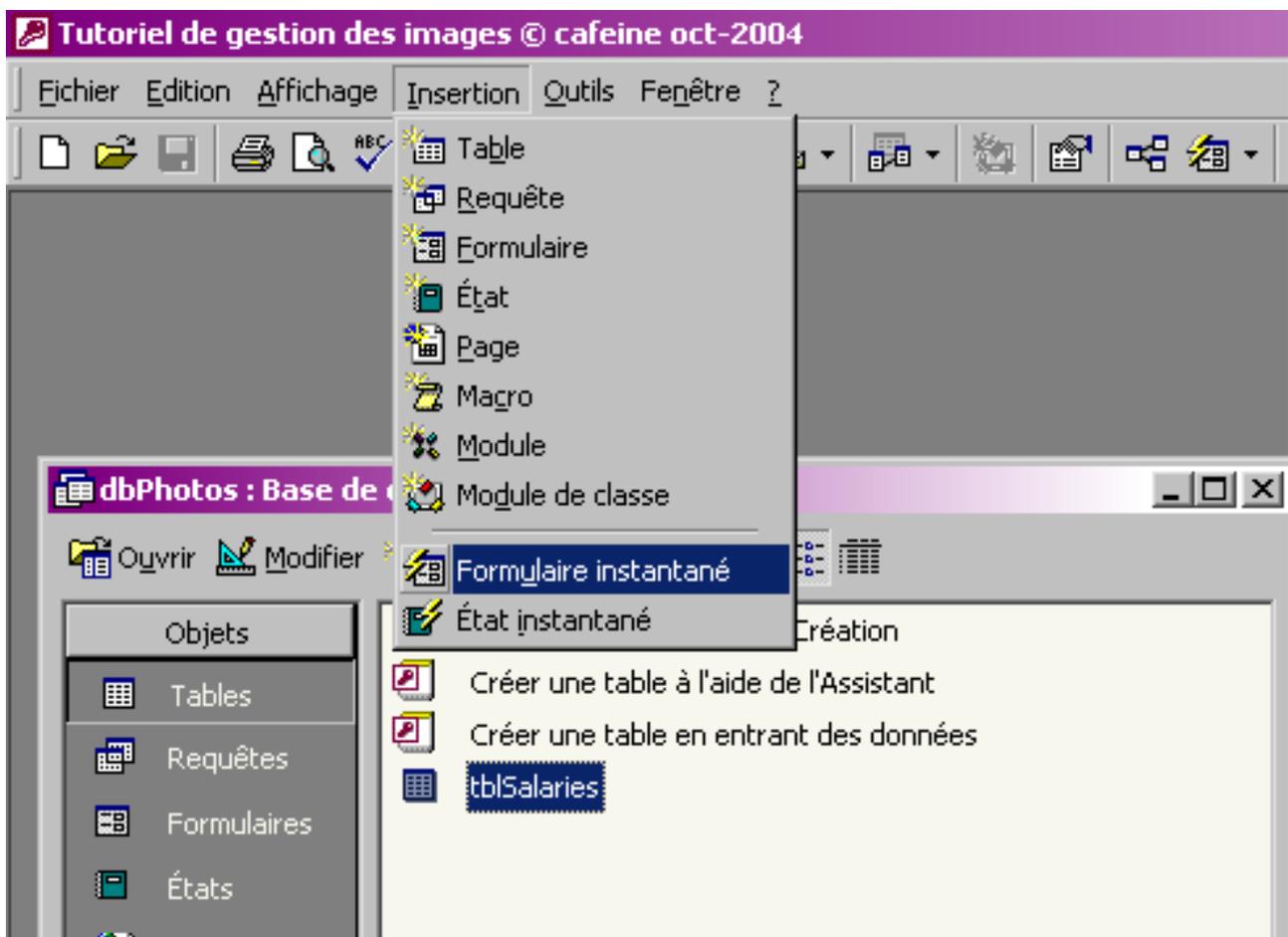
4- Conception du formulaire

a. Génération automatique

La construction du formulaire sera facilitée par l'assistant formulaire instantané.

Il suffira pour cela de sélectionner la table **tblSalaries** dans la fenêtre de base de données et de cliquer sur l'icône "Formulaire instantané"

ou MENU > Insertion > Formulaire instantané.

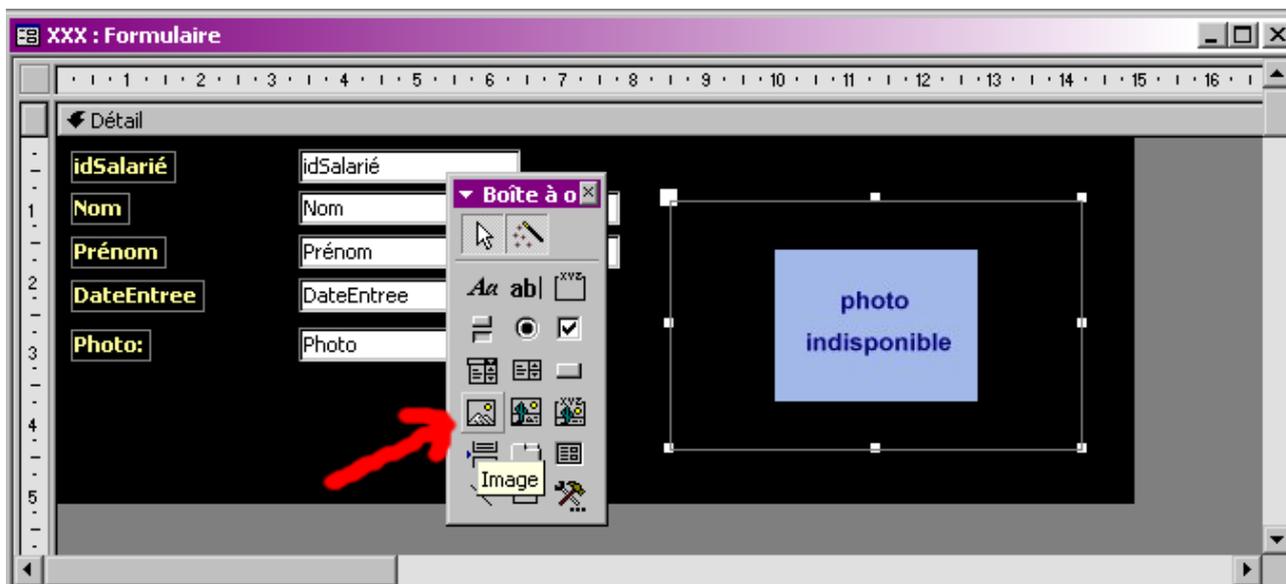


Par la génération automatique, le formulaire est lié à la table **tblSalaries**

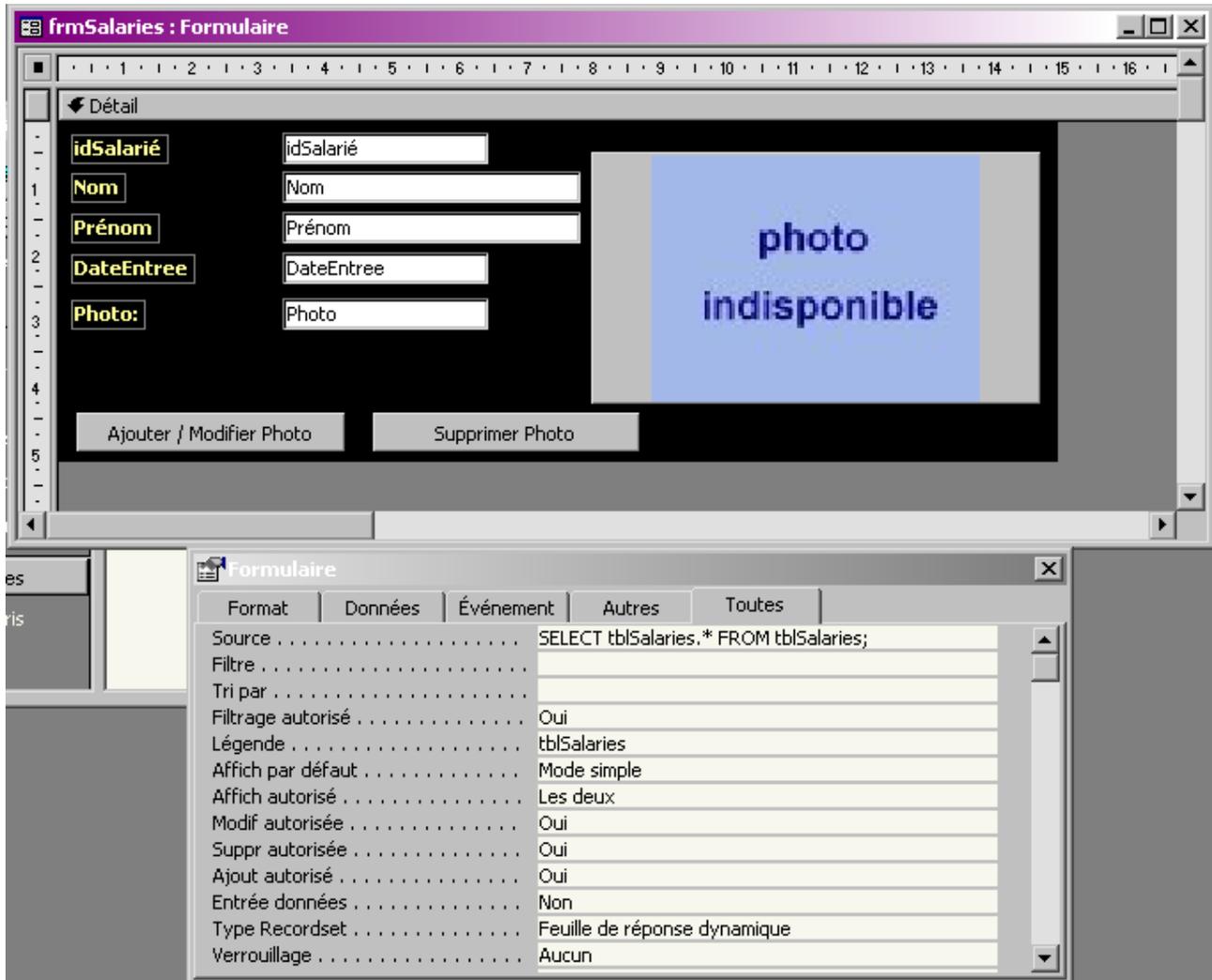
b. Ajout des contrôles

Il nous reste à placer le contrôle image qui affichera les photos.

Puis à créer deux boutons de commandes : un pour ajouter / modifier et l'autre pour effacer la photo.



Le résultat final doit donner ceci :



5- Codage du formulaire

a. Affichage des photos

Nous allons coder en premier lieu l'affichage automatique de l'image.

```
Private Sub Form_Current()
' L'événement Activation (Current) se produit lorsque le focus passe à un enregistrement
' donné pour en faire l'enregistrement en cours, ou lorsque le formulaire est
' Actualisé ou en Actualisation.

' si le nom du salarié est non vide : on visualise un enregistrement
' sinon cela indique que nous sommes sur un enregistrement vierge, donc en cours de saisie.
' Me.Caption : gère le titre du formulaire.
If Len(Me.Nom) > 0 Then
    Me.Caption = "Détails pour le salarié : " & Me.Nom & " - " & Me.Prénom
Else
    Me.Caption = "Saisie d'un nouveau salarié"
End If

' Gestion des erreurs
On Error GoTo Catch02

' si la photo n'est pas définie, on affiche la photo blank.jpg
' CurrentProject.Path : est le chemin de l'application
If Len(Me.Photo) > 0 Then
    Me.imgPhoto.Picture = Me.Photo
Else
    Me.imgPhoto.Picture = CurrentProject.Path & "\images\blank.jpg"
End If

DisplayPhoto

Exit Sub

Catch02:
Select Case Err.Number
    Case 2114
        'Cas d'un type de fichier photo non supporté ...
        MsgBox "Le format de l'image n'est supporté par le contrôle image Picture", vbCritical +
vbOKOnly, "Application Photos"
        Me.imgPhoto.Picture = CurrentProject.Path & "\images\blank.jpg"
        Me.Photo = vbNullString
    Case 2220
        'Cas d'un emplacement non valide du fichier image
        MsgBox "Le fichier image n'a pas été trouvé à l'emplacement indiqué : " & vbCrLf & _
            Me.Photo, vbCritical + vbOKOnly, "Application Photos"
        Me.imgPhoto.Picture = CurrentProject.Path & "\images\blank.jpg"
        Me.Photo = vbNullString
    Case Else
        ' tout autre cas d'erreur
        MsgBox "Erreur inattendue : " & Err.Number & vbCrLf & Err.Description, vbCritical +
vbOKOnly, "Application Photos"
End Select
Err.Clear

End Sub
```

Me.imgPhoto.Picture = Me.Photo : la propriété **.Picture** attribue le chemin physique de la photo au contrôle image, ce qui a pour effet son affichage immédiat.

Le chemin physique a été stocké dans la base via le champ **Photo** qui est dans le contrôle **Photo**.

DisplayPhoto est la sub d'affichage des photos dans le contrôle Image appelé : **imgPhoto**.

Cette procédure ajuste l'affichage de la photo au cas où ses dimensions dépasseraient celles du contrôle photo.

Me.imgPhoto.ImageHeight : donne la hauteur de la Photo (Fichier .jpg par exemple)

Me.imgPhoto.Height : donne la hauteur du contrôle d'affichage.

```
Sub DisplayPhoto()
' Traitement en fonction de la taille de l'image

' regarde si la hauteur de l'image dépasse celle du controle Picture
If Me.imgPhoto.ImageHeight > Me.imgPhoto.Height Then
' met le controle en mode zoom
Me.imgPhoto.SizeMode = 3
Else
' met le contrôle en mode respect de la taille originale
Me.imgPhoto.SizeMode = 0
End If

' si la largeur dépasse et qu'on est en mode taille réelle ...
If (Me.imgPhoto.ImageWidth > Me.imgPhoto.Width) And (Me.imgPhoto.SizeMode) = 0 Then
' on met en mode zoom
Me.imgPhoto.SizeMode = 3
End If

End Sub
```

b. Ajout - Suppression des photos

Nous créons les deux sub d'événement Clic de nos boutons de commande nommés :

. **cmdDelete** : bouton d'effacement de la photo

. **cmdPhoto** : bouton d'ajout / modification de la photo

```
Private Sub cmdDelete_Click()
' Bouton de commande d'effacement de la photo

' supprime l'adresse de la photo
Me.Photo = vbNullString

' affiche l'image blank.jpg
Me.imgPhoto.Picture = CurrentProject.Path & "\images\blank.jpg"

' redimensionne la photo
DisplayPhoto

End Sub

Private Sub cmdPhoto_Click()
' Bouton d'ajout - modification de photo
Dim strLink As String

' Gestion des erreurs
On Error GoTo Catch01

' récupération du chemin physique de la photo
' par la boîte de dialogue
strLink = OuvrirUnFichier(Me.Hwnd, _
" Sélectionner une photo pour le salarié " & Me.Nom, _
1)

' si la boîte renvoie une adresse non nulle
```

```
If Len(strLink) > 0 Then
    ' tentative d'affichage de la photo
    Me.imgPhoto.Picture = strLink
    Me.Photo = strLink
End If

DisplayPhoto
Exit Sub

Catch01:
Select Case Err.Number
    Case 2114
        'Cas d'un type de fichier photo non supporté ...
        ' on sort de la procédure
        MsgBox "Le format de l'image n'est supporté par le contrôle image Picture", vbCritical +
vbOKOnly, "Application Photos"
        Exit Sub
    Case 2220
        'Cas d'un emplacement non valide du fichier image
        MsgBox "Le fichier image n'a pas été trouvé à l'emplacement indiqué : " & vbCrLf & _
            Me.Photo, vbCritical + vbOKOnly, "Application Photos"
        Exit Sub
    Case Else
        ' tout autre cas d'erreur
        MsgBox "Erreur inattendue : " & Err.Number & vbCrLf & Err.Description, vbCritical +
vbOKOnly, "Application Photos"
End Select
Err.Clear

End Sub
```

strLink = OuvrirUnFichier(...

Attribue à la chaîne **strLink** le chemin renvoyé par la boite de dialogue de pointage de fichier.

On ne reviendra pas sur le codage de la fonction **OuvrirUnFichier()** qui est disponible dans la FAQ : [ici](#)

6- Conclusion

J'espère que ce court tutoriel vous aura permis de vous familiariser avec la gestion des images externes dans MS Access.

Vous pourrez trouver la base exemple et les photos qui l'accompagnent dans le fichier .zip suivant : [Base exemple du tutoriel \[99 ko\]](#)