

Créer un organigramme et un trombinoscope dans Excel

par [SilkyRoad \(silkyroad.developpez.com\)](http://silkyroad.developpez.com)

Date de publication : 11/03/2007

Dernière mise à jour : 12/05/2007

Cette page montre comment créer un organigramme et un trombinoscope dans Excel.

Testé avec Excel2002/Windows XP.


- I - Introduction
- II - Description
- III - Les procédures
- IV - Téléchargement

I - Introduction

Ce tutoriel propose un exemple d'organigramme et un trombinoscope dans Excel.

Vous pourrez visualiser dans un UserForm la structure organisationnelle d'une équipe de travail ou d'une association, l'organigramme d'une entité... et aussi retrouver le nom d'une personne à partir de sa photo: Le classeur disposant d'une option pour afficher toutes les images jpg du répertoire sous forme de planche contact.

Cette présentation est avant tout un support de cours pour:

- * Utiliser un TreeView.
- * Créer une page html dynamiquement.
- * Visualiser la page html dans un WebBrowser ( **Navigateur Web Microsoft**).
- * Utiliser un module de classe pour gérer le contenu du WebBrowser (Identification de l'évènement Clic sur les différentes images de la page html).

II - Description

Pour commencer, vous devez mettre en place les informations nécessaires à la création de l'organigramme, dans l'onglet **Structure**.

Chaque colonne (de **B** à **M**) définit un niveau d'arborescence dans le TreeView. Les numéros de téléphone et de fax sont stockés dans les colonnes **O** et **P**.

La croix "x" dans la colonne **N** sert à différencier les fonctions (Responsable d'exploitation, administratif, fabrication...) et le nom des personnes.

Structure organisationnelle établissement			
Responsable d'exploitation		tel	fax
Nom01 prenom01	(x) Sert à identifier les personnes (et séparer les titres de fonctions)	x 8551	8570
Administratif			
Nom02 prenom02		x 8500	8570
Nom03 prenom03		x 8507	8570
Responsable Fabrications			
Nom04 prenom04		x 8530	8537
Procédés			
Nom05 prenom05		x	8537
Fabrication			
Nom06 prenom06		x 8531	8537
Nom07 prenom07		x 8532	8537
Nom08 prenom08		x 8533	8537

Le nom et le prénom sont volontairement dans la même cellule afin de limiter les risques d'homonymes, et simplifier la gestion des images.

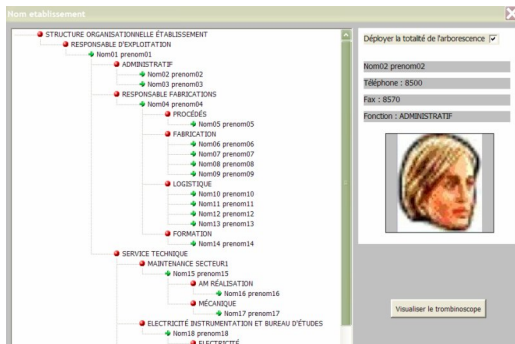
Vous pourrez bien entendu modifier la démo en fonction de vos préférences.

Les images (non obligatoires pour faire fonctionner le classeur) doivent être placées dans le même répertoire que le classeur.

Les photos doivent être nommées de manière identique à l'onglet **Structure**:

Par exemple [Nom01 prenom01.jpg](#)

Un contrôle **TreeView** est utilisé pour visualiser l'organigramme.



Lors de l'initialisation du UserForm, la procédure boucle sur les cellules de la feuille **Structure** afin de mettre en place l'arborescence du TreeView.

Vba

```
'Boucle sur les éléments de la structure pour remplir le TreeView
For Each Cell In Feuil2.Range("A1:A" & Feuil2.Range("N65533").End(xlUp).Row)
    NumCol = Cell.End(xlToRight).Column
    NumLig = Cell.Row

    'Les informations de la colonne B correspondent à un noeud principal.
    If NumCol = 2 Then
        TreeView1.Nodes.Add , , "maClé" & NumLig & NumCol, _
            UCase(Feuil2.Cells(NumLig, NumCol)), "Img1", "Img1"
    Else
        k = Feuil2.Cells(NumLig, NumCol).Offset(0, -1).End(xlUp).Row
        j = Feuil2.Cells(NumLig, NumCol).Offset(0, -1).Column

        'S'il s'agit d'un membre de l'équipe:
        '(Dans ce cas la colonne N contient la lettre "x")
        If Feuil2.Cells(NumLig, 14) = "x" Then
            TreeView1.Nodes.Add _
                "maClé" & k & j, tvwChild, "maClé" & NumLig & NumCol, _
                    Feuil2.Cells(NumLig, NumCol), "Img2", "Img2"
        Else
            'S'il s'agit d'un titre de service:
            TreeView1.Nodes.Add _
                "maClé" & k & j, tvwChild, "maClé" & NumLig & NumCol, _
                    UCase(Feuil2.Cells(NumLig, NumCol)), "Img1", "Img1"
        End If
    End If
Next Cell
```

Par défaut L'arborescence est fermée.

Cochez la CheckBox "**Déployer la totalité de l'arborescence**" pour afficher toute la structure. Un double clic sur chaque noeud déploie le niveau inférieur.

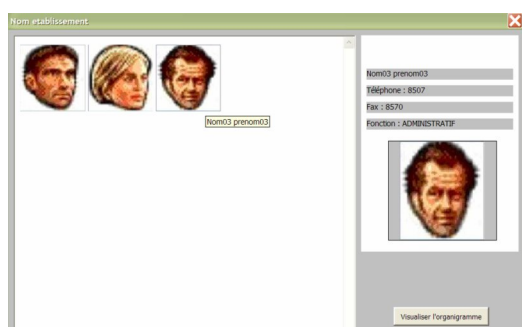
Le contrôle **ImageList** permet d'afficher une petite image sur les nœuds du TreeView: un point rouge pour un titre de fonction, une flèche verte pour le nom des personnes.

Dès que vous cliquez sur un nom dans le TreeView, les informations associées à la personne s'affichent dans l'UserForm:

- * Nom Prénom
- * Numéro de téléphone
- * Fonction
- * La photo (si elle existe dans le même répertoire que le classeur).

Ce n'est qu'un exemple et vous pourrez par la suite adapter les champs en fonction de vos besoins.

Un contrôle **WebBrowser** est utilisé comme support pour le trombinoscope.



Lorsque vous cliquez sur le bouton "**Visualiser le trombinoscope**", une page html est créée dynamiquement, basée sur les photos disponibles dans le répertoire:

Vba

'Répertoire contenant le classeur

Vba

```
chemin = ThisWorkbook.Path
'Recherche des images jpg dans le repertoire
Fichier = Dir(chemin & "\*.jpg")

'Création d'une page html qui s'affichera dans le WebBrowser
Open ThisWorkbook.Path & "\browserImage.html" For Output As #1
Print #1, "<HTML>"
Print #1, "<HEAD>"
Print #1, "<TITLE>" & chemin & "</TITLE>"

Do
    S = chemin & "\" & Fichier
    ProprietesImages = Left(Fichier, Len(Fichier) - 4)

    'création vignette
    X = "<A><IMG WIDTH=120 HEIGHT=120 SRC='" & S & _
        "'ALT='" & ProprietesImages & "'></IMG></A>"
    'création vignette et lien hypertexte pour chaque image
    'X = "<A href='" & S & "'><IMG WIDTH=90 HEIGHT=90 SRC='" & S & _
        "'ALT='" & ProprietesImages & "'></IMG></A>"
    Print #1, X

    Fichier = Dir
Loop Until Fichier = ""

Close #1

'Affiche la page html dans le WebBrowser.
WebBrowser1.Navigate ThisWorkbook.Path & "\browserImage.html"
```

Le TreeView est masqué pour que le WebBrowser soit placé au premier plan.

Les images sont présentées sous forme de vignettes. Une infobulle est aussi créée dans la page html, pour afficher le nom et le prénom de la personne lorsque le curseur de la souris passe sur l'image. Ensuite, un clic sur l'image permet de récupérer les informations complètes au sujet de la personne sélectionnée (Nom, Prénom, Numéro de téléphone, Fonction).

Remarque:

La page html ne gère pas les caractères spéciaux dans le nom des fichiers images (les apostrophes par exemple).

L'évènement **Clic** sur les images est géré par un module de classe.

Toutes les images sont d'abord intégrées dans la classe lorsque la page html est totalement chargée dans le WebBrowser:

Vba

```
Private Sub WebBrowser1_DocumentComplete(ByVal pDisp As Object, URL As Variant)
    Dim Cl As Classe1
    Dim i As Integer
    Dim imgHtml As HTMLImg

    Set Collect = New Collection
    Set maPageHtml = WebBrowser1.Document

    'Boucle sur les images contenues dans le WebBrowser
    For i = 0 To maPageHtml.images.Length - 1
        Set imgHtml = maPageHtml.images.Item(i)

        'ajoute l'objet dans la classe
        Set Cl = New Classe1
        Set Cl.Imge = imgHtml
        Collect.Add Cl
    Next i

End Sub
```

Ensuite, le module de classe va pouvoir gérer l'évènement **onclick** sur chaque image de la page html:

Vba

```
'A placer dans un module de classe nommé "Classe1"
'
Option Explicit

'Nécessite d'activer la référence "Microsoft HTML Object Library"
Public WithEvents Imge As MSHTML.HTMLImg

'Exemple pour gérer l'évènement clic sur les objets type MSHTML.HTMLImg (images)
'dans le WebBrowser.
Private Function Imge_onclick() As Boolean
    Dim Cible As String, Fichier As String
    Dim m As Integer

    Cible = Imge.alt

    For m = 1 To UserForm1.TreeView1.Nodes.Count
        If Cible = UserForm1.TreeView1.Nodes.Item(m).Text Then

            UserForm1.Label2 = UserForm1.TreeView1.Nodes.Item(m).Text
            UserForm1.Label3 = "Téléphone : " & Feuil2.Cells(m, 15)
            UserForm1.Label4 = "Fax : " & Feuil2.Cells(m, 16)
            UserForm1.Label5 = "Fonction : " & UserForm1.TreeView1.Nodes.Item(m).Parent

            Fichier = ThisWorkbook.Path & "\" & Cible & ".jpg"

            If Dir(Fichier) <> "" Then
                UserForm1.Image1.Picture = LoadPicture(Fichier)
            Else
                Set UserForm1.Image1.Picture = Nothing
            End If
        End If
    Next m
End Function
```

La procédure nécessite d'activer la référence [Microsoft HTML Object Library](#).

Dans l'éditeur de macros (Alt+F11):

Menu

Outils

Références

Cochez la ligne "Microsoft HTML Object Library"

Cliquez sur le bouton OK pour valider.

Cette référence (ou bibliothèque) sert à piloter tous les types d'objets contenus dans une page html.

Cliquez sur le bouton "**Visualiser l'organigramme**" pour réafficher le TreeView.

III - Les procédures

Dans un module standard:

Vba

```
'-----  
'A placer dans un module standard  
Option Explicit  
  
Public Collect As Collection  
'-----  
  
Sub Lancer()  
    UserForm1.Show  
End Sub
```

Dans le module objet du UserForm:

Vba

```
Option Explicit  
Option Compare Text  
  
Dim maPageHtml As HTMLDocument  
  
Private Sub UserForm_Initialize()  
    Dim NumCol As Integer, j As Integer  
    Dim NumLig As Integer, k As Integer  
    Dim Cell As Range  
    Dim Image1 As String, Image2 As String  
  
    '--- Spécifie les images qui s'affichent dans les noeuds.  
    'Les images doivent être dans le même répertoire que le classeur.  
    Image1 = ThisWorkbook.Path & "\redball.gif"  
    Image2 = ThisWorkbook.Path & "\grnarrow.gif"  
  
    'Supprime le contenu de l'ImageList  
    Me.ImageList1.ListImages.Clear  
  
    'chargement des images  
    Me.ImageList1.ListImages.Add 1, "Img1", LoadPicture(Image1)  
    Me.ImageList1.ListImages.Add 2, "Img2", LoadPicture(Image2)  
    'Associe les images au TreeView  
    Set Me.TreeView1.ImageList = Me.ImageList1  
    '---  
  
    'Boucle sur les éléments de la structure pour remplir le TreeView  
    For Each Cell In Feuil2.Range("A1:A" & Feuil2.Range("N65533").End(xlUp).Row)  
        NumCol = Cell.End(xlToRight).Column  
        NumLig = Cell.Row
```

Vba

```

If NumCol = 2 Then
    TreeView1.Nodes.Add , , "maClé" & NumLig & NumCol, _
        UCase(Feuil2.Cells(NumLig, NumCol)), "Img1", "Img1"
Else
    k = Feuil2.Cells(NumLig, NumCol).Offset(0, -1).End(xlUp).Row
    j = Feuil2.Cells(NumLig, NumCol).Offset(0, -1).Column

    'S'il s'agit d'un membre de l'équipe:
    '(Dans ce cas la colonne N contient la lettre "x")
    If Feuil2.Cells(NumLig, 14) = "x" Then
        TreeView1.Nodes.Add _
            "maClé" & k & j, tvwChild, "maClé" & NumLig & NumCol, _
                Feuil2.Cells(NumLig, NumCol), "Img2", "Img2"
        Else
            'S'il s'agit d'un titre de service:
            TreeView1.Nodes.Add _
                "maClé" & k & j, tvwChild, "maClé" & NumLig & NumCol, _
                    UCase(Feuil2.Cells(NumLig, NumCol)), "Img1", "Img1"
        End If
    End If
Next Cell

    TreeView1.Style = 5
End Sub

Private Sub UserForm_Activate()
    'Pour afficher l'UserForm en plein écran

    'With Me
        .StartupPosition = 3
        .Width = Application.Width
        .Height = Application.Height
        .Left = 0
        .Top = 0
    'End With
End Sub

'Déploie l'ensemble du TreeView si la checkBox
'"Déployer la totalité de l'arborescence" est cochée.
Private Sub CheckBox1_Click()
    Dim i As Byte

    If CheckBox1 Then
        'Boucle sur tous les noeuds du TreeView.
        For i = 1 To TreeView1.Nodes.Count
            TreeView1.Nodes.Item(i).Expanded = True
        Next
    Else
        For i = 1 To TreeView1.Nodes.Count
            TreeView1.Nodes.Item(i).Expanded = False
        Next
    End If

    'Positionne le 1er noeud dans la partie visible du TreeView
    TreeView1.Nodes.Item(1).EnsureVisible
End Sub

'Evenement Clic sur un élément du treeView.
Private Sub TreeView1_Click()

```

Vba

```

Dim leNom As String, Fichier As String

'Vérifie si l'élément sélectionné correspond à une personne ou à un titre
'de service.
'(La colonne N contient la lettre "x" s'il s'agit d'une personne)
If Feuil2.Cells(TreeView1.SelectedItem.Index, 14) <> "" Then
    'Affiche les informations sur la personne sélectionnée.
    Label2 = TreeView1.SelectedItem.Text
    Label3 = "Téléphone : " & Feuil2.Cells(TreeView1.SelectedItem.Index, 15)
    Label4 = "Fax : " & Feuil2.Cells(TreeView1.SelectedItem.Index, 16)
    Label5 = "Fonction : " & TreeView1.SelectedItem.Parent

    leNom = TreeView1.SelectedItem.Text
    'Définit l'image associée au nom sélectionné.
    Fichier = ThisWorkbook.Path & "\" & leNom & ".jpg"

    'Vérifie si le fichier image existe dans le répertoire
    If Dir(Fichier) <> "" Then
        'Charge l'image si elle existe.
        Image1.Picture = LoadPicture(Fichier)
    Else
        'Sinon fait le ménage dans le contrôle Image
        Set Image1.Picture = Nothing
    End If
End If

End Sub

'Affichage du trombinoscope:
'(Création d'une planche contact pour visualiser les images dans le WebBrowser)
Private Sub CommandButton1_Click()
    Dim Fichier As String
    Dim S As String, X As String, chemin As String
    Dim ProprietesImages As String

    If WebBrowser1.Visible = True Then
        WebBrowser1.Visible = False
        Label1.Visible = True
        CheckBox1.Visible = True
        CommandButton1.Caption = "Visualiser le trombinoscope"
    Exit Sub
    End If

    Label1.Visible = False
    CheckBox1.Visible = False
    WebBrowser1.Visible = True
    CommandButton1.Caption = "Visualiser l'organigramme"

    'Répertoire contenant le classeur
    chemin = ThisWorkbook.Path
    'Recherche des images jpg dans le repertoire
    Fichier = Dir(chemin & "\*.jpg")

    'Création d'une page html qui s'affichera dans le WebBrowser
    Open ThisWorkbook.Path & "\browserImage.html" For Output As #1
        Print #1, "<HTML>"
        Print #1, "<HEAD>"
        Print #1, "<TITLE>" & chemin & "</TITLE>"

    Do
        S = chemin & "\" & Fichier
        ProprietesImages = Left(Fichier, Len(Fichier) - 4)

        'création vignette
        X = "<A><IMG WIDTH=120 HEIGHT=120 SRC='" & S & _

```

Vba

```

        "'ALT='" & ProprietesImages & "'></IMG></A>"
        'création vignette et lien hypertexte pour chaque image
        'X = "<A href='" & S & "'><IMG WIDTH=90 HEIGHT=90 SRC='" & S & _
        "'ALT='" & ProprietesImages & "'></IMG></A>"
        Print #1, X

        Fichier = Dir
        Loop Until Fichier = ""

    Close #1

    'Affiche la page html dans le WebBrowser.
    WebBrowser1.Navigate ThisWorkbook.Path & "\browserImage.html"

End Sub

Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)
    'Suppression de la page html (si elle existe) lors de la fermeture de l'USF
    If Dir(ThisWorkbook.Path & "\browserImage.html") <> "" Then _
        Kill ThisWorkbook.Path & "\browserImage.html"
End Sub

'Cet évènement est déclenché lorsqu'une page est totalement chargée dans le WebBrowser:
'Dans cet exemple, toutes les images de la page html sont prises en compte dans
'le module de classe dès que la page est chargée.
Private Sub WebBrowser1_DocumentComplete(ByVal pDisp As Object, URL As Variant)
    Dim Cl As Class1
    Dim i As Integer
    Dim imgHtml As HTMLImg

    Set Collect = New Collection
    Set maPageHtml = WebBrowser1.Document

    'Boucle sur les images contenues dans le WebBrowser
    For i = 0 To maPageHtml.images.Length - 1
        Set imgHtml = maPageHtml.images.Item(i)

        'ajoute l'objet dans la classe
        Set Cl = New Class1
        Set Cl.Imge = imgHtml
        Collect.Add Cl
    Next i

End Sub

Private Sub WebBrowser1_BeforeNavigate2(ByVal pDisp As Object, _
    URL As Variant, Flags As Variant, TargetFrameName As Variant, _
    postData As Variant, Headers As Variant, Cancel As Boolean)

    'Fait le ménage avant d'afficher une nouvelle page
    Set Collect = Nothing
    Set maPageHtml = Nothing

End Sub

```

Dans un module de classe nommé **Classe1**:

Vba

```
'-----  
'A placer dans un module de classe nommé "Classe1"  
'  
Option Explicit  
  
'Nécessite d'activer la référence "Microsoft HTML Object Library"  
Public WithEvents Imge As MSHTML.HTMLImg  
  
'Exemple pour gérer l'évènement clic sur les objets type MSHTML.HTMLImg (images)  
'dans le WebBrowser.  
Private Function Imge_onclick() As Boolean  
    Dim Cible As String, Fichier As String  
    Dim m As Integer  
  
    Cible = Imge.alt  
  
    For m = 1 To UserForm1.TreeView1.Nodes.Count  
        If Cible = UserForm1.TreeView1.Nodes.Item(m).Text Then  
  
            UserForm1.Label2 = UserForm1.TreeView1.Nodes.Item(m).Text  
            UserForm1.Label3 = "Téléphone : " & Feuil2.Cells(m, 15)  
            UserForm1.Label4 = "Fax : " & Feuil2.Cells(m, 16)  
            UserForm1.Label5 = "Fonction : " & UserForm1.TreeView1.Nodes.Item(m).Parent  
  
            Fichier = ThisWorkbook.Path & "\" & Cible & ".jpg"  
  
            If Dir(Fichier) <> "" Then  
                UserForm1.Image1.Picture = LoadPicture(Fichier)  
            Else  
                Set UserForm1.Image1.Picture = Nothing  
            End If  
        End If  
    Next m  
End Function  
'-----
```

IV - Téléchargement

Téléchargez le classeur démo.

