

Gérer la navigation dans un diaporama PowerPoint

Par Philippe JOCHMANS 

Date de publication : 19 mars 2008

Dernière mise à jour : 19 mars 2008

Cet article a pour but de vous montrer comment gérer la navigation dans un diaporama PowerPoint.

I - INTRODUCTION.....	3
II - NAVIGATION AUTOMATIQUE.....	3
II-A - Paramétrage manuel.....	4
II-B - Paramétrage par le code.....	4
III - LES BOUTONS D'ACTION.....	5
III-A - Détail des boutons d'action.....	5
III-B - Affecter les boutons d'action à toutes les diapositives.....	7
III-B-1 - Manuellement.....	7
III-B-2 - Par programmation.....	9
III-C - Formatage des boutons d'action.....	12
IV - LES IMAGES ET LES FORMES.....	12
IV-A - Manuellement.....	12
IV-A-1 - Création des formes.....	12
IV-A-2 - Les actions sur les formes.....	13
IV-B - Par programmation.....	14
IV-C - Cas particulier des images.....	16
V - LES LIENS HYPERTEXTES.....	16
VI - CONCLUSION.....	16
VII - REMERCIEMENTS.....	17

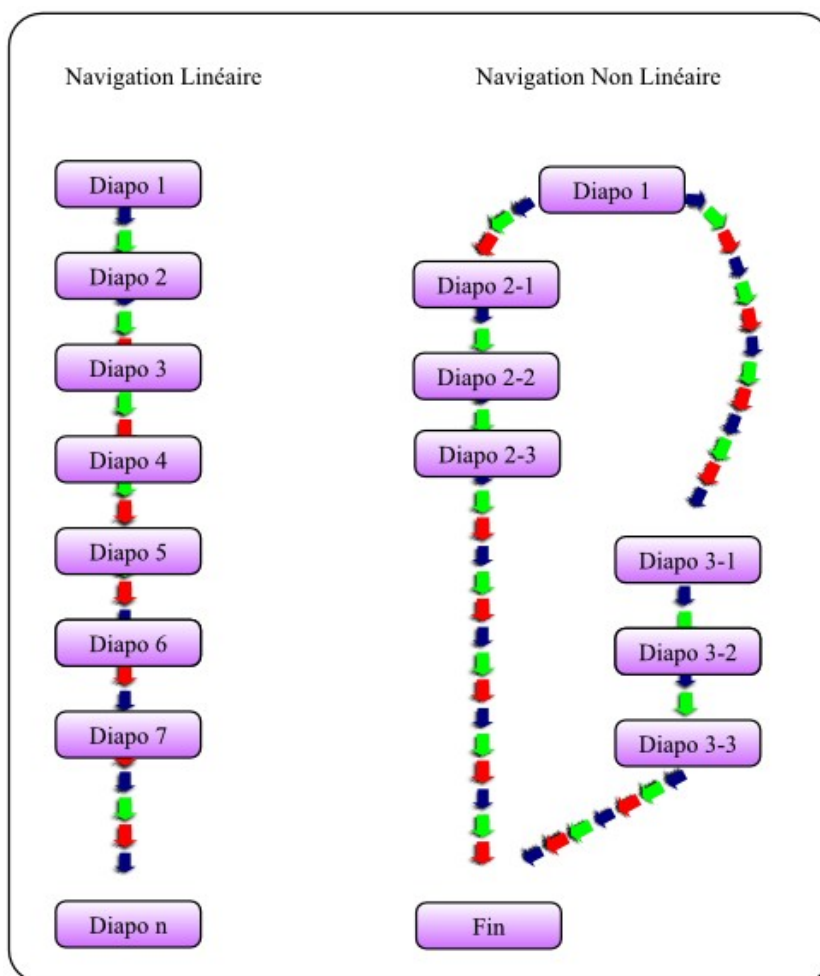
I - INTRODUCTION

Le but de cet article est de vous montrer comment on peut naviguer entre les diapositives dans PowerPoint.

Nous pouvons distinguer deux types de navigations :

- **la navigation linéaire** : on commence par la première diapositive, et on finit par la dernière ;
- **une navigation non linéaire** : l'ensemble des diapositives est composé en bloc et on passe d'un bloc à l'autre.

Le schéma ci-dessous illustre cela.



II - NAVIGATION AUTOMATIQUE

Par défaut, la transition d'une diapositive à l'autre se fait lors du clic sur le bouton de la souris.

Cependant, cette méthode a pour inconvénients :

- il faut que toutes les animations de la diapositive se soient déroulées pour le passage à la diapositive suivante ;
- votre diaporama est linéaire et vous ne pouvez imposer le passage vers une diapositive qui ne se trouve pas dans la suite logique ;

- l'inconvénient également d'une navigation automatique est que le timing entre les slides (diapositives) doit être impeccable, le lecteur doit avoir suffisamment de temps pour lire les infos mais le slide ne doit pas être trop long non plus.

II-A - Paramétrage manuel

Le paramétrage par défaut est heureusement modifiable.

- Affichez le volet Office (Menu Affichage/Volet Office).
- Sélectionnez transition.

Vous trouverez en bas de celui-ci l'image suivante :

- Manuellement : si cette case est cochée, cela indique que la transition entre les diapositives se fera manuellement, sinon, il faudra gérer le passage d'une diapositive à l'autre.
- Automatiquement après : permet de déterminer une temporisation entre chaque diapositive.
- Appliquer à toutes les diapositives : cela vous permet d'affecter le choix à l'ensemble de votre diaporama ou non.

Dans la suite de cet article nous allons décocher la case **Manuellement**, car nous voulons gérer nous-même le passage d'une diapositive à l'autre.

PowerPoint 2007

Pour PowerPoint 2007, allez dans le menu animation pour voir apparaître le ruban animation, vous trouverez dans celui-ci les mêmes paramètres.



II-B - Paramétrage par le code

Il est possible de programmer l'annulation de la transition manuelle entre les diapositives.

Le code suivant permet de désactiver la transition manuelle dans un diaporama.

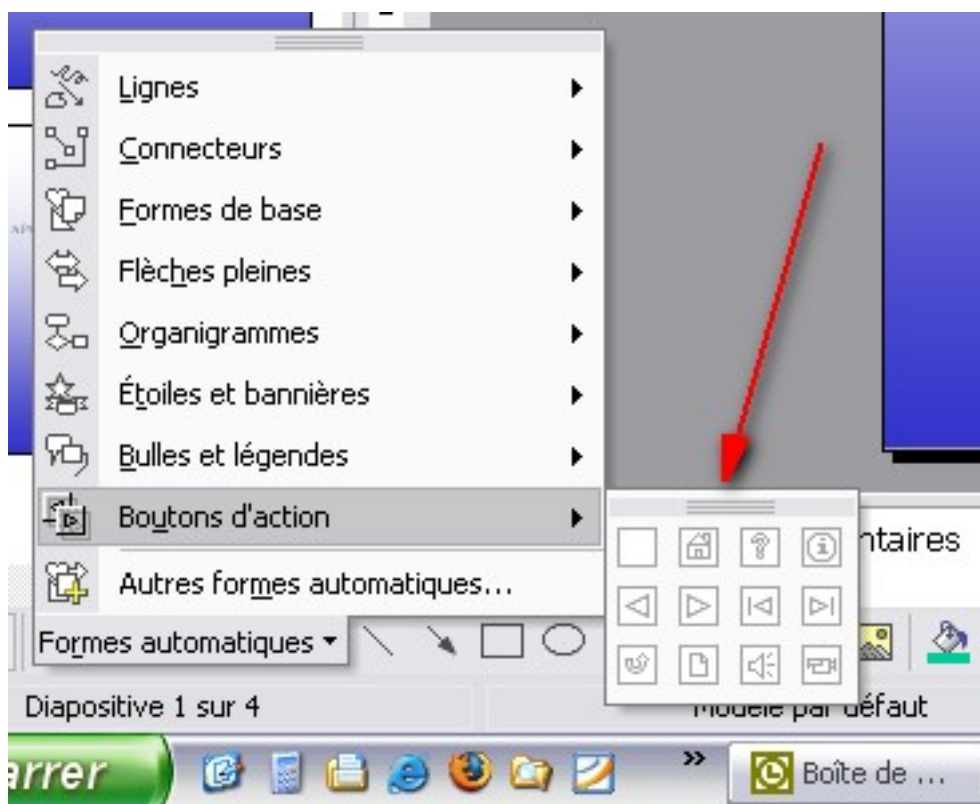
VBA

```
Dim sld As Slide
For Each sld In ActivePresentation.Slides
    sld.SlideShowTransition.AdvanceOnClick = msoFalse
Next sld
```

III - LES BOUTONS D'ACTION

Les boutons d'action sont des boutons prédéfinis qui vont vous permettre de gérer la navigation entre vos diapositives.

Ceux-ci se trouvent dans la barre d'outils dessin sous PPT 2002 et 2003.



Vous pouvez également utiliser le menu Diaporama/Boutons d'action.

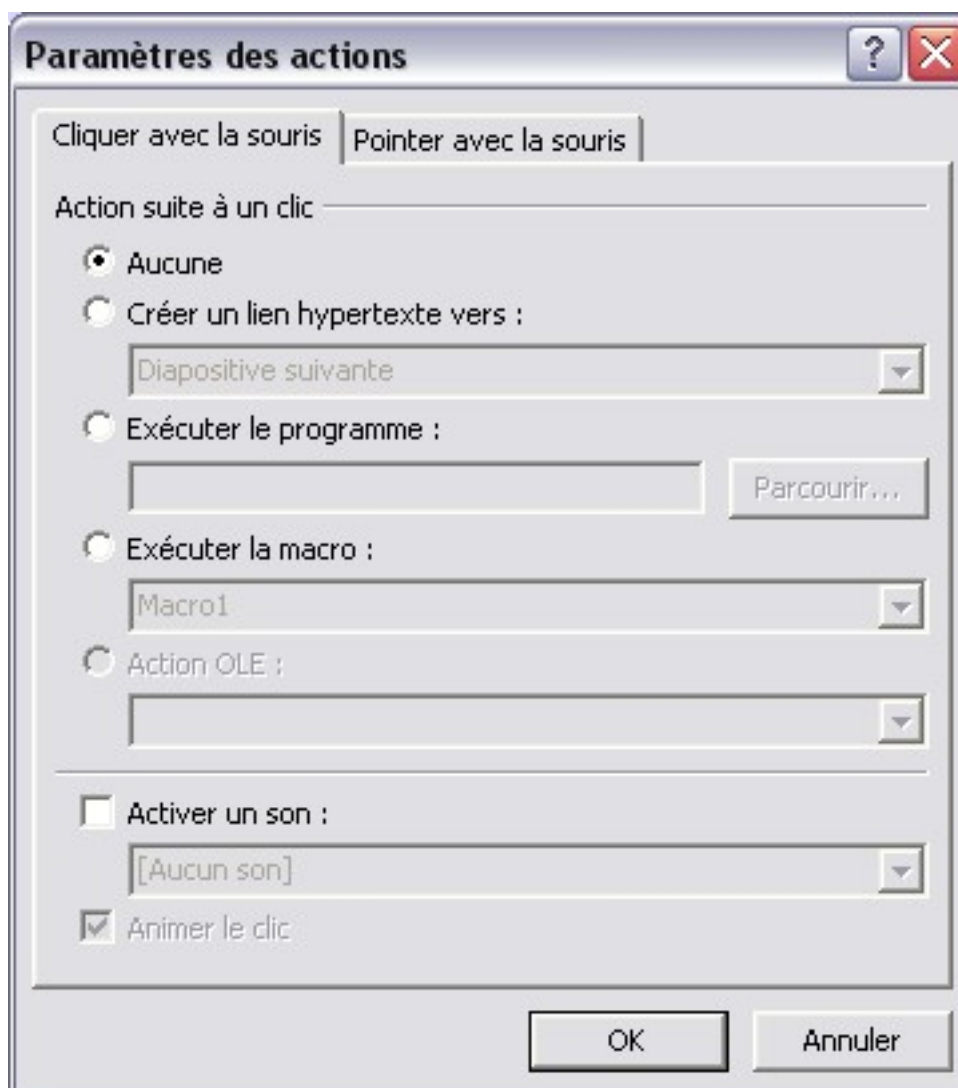
Sous PPT 2007 :

Activer le menu Insertion, dans le ruban sélectionner Formes, et ceux-ci se trouvent en bas de la liste déroulante.

III-A - Détail des boutons d'action

PowerPoint donne la possibilité d'utiliser 12 boutons d'actions.

Lorsque vous positionnez un bouton d'action sur une diapositive une fenêtre de paramétrage s'ouvre.



Cette fenêtre va vous permettre de paramétrer l'action du bouton, cependant, certains sont déjà préparamétrés.

Il peut y avoir deux types d'actions : sur le clic du bouton ou sur le survol par le curseur de la souris (vous trouverez le même type d'action pour ces deux évènements).

- **Aucune** : aucune action ne sera effectuée
- **Créer un lien hypertexte vers** : vous permettra de créer un lien hypertexte vers une autre diapositive ou une URL. (Pour certains boutons d'action, cette propriété est définie par défaut.)
- **Exécuter le programme** : va vous permettre de lancer un programme (.exe).
- **Exécuter la macro** : va vous permettre d'exécuter une macro VBA.
- **Action Ole** : n'est pas utilisée pour les boutons d'action.
- **Activer un son** : va vous permettre de lancer un son lors d'une action sur le bouton (le son doit d'abord être inséré dans la diapositive pour apparaître dans la liste).

i Vous pourrez ouvrir cette fenêtre à tout moment en effectuant un clic droit sur la souris (menu contextuel) et en sélectionnant Paramètres des Actions.

Le tableau suivant vous présentera les 12 boutons d'action.

Bouton d'Action	Image	Description
Bouton Personnalisé		Ce bouton va vous permettre de personnaliser une action grâce à la fenêtre de paramétrage des actions.

		<p>Ce bouton ne comportant pas d'icône, il est possible d'écrire du texte à l'intérieur, pour cela afficher le menu contextuel (clic droit), et sélectionnez Modifier Texte.</p> <p>Pour que le texte s'adapte au bouton ou l'inverse, dans le menu contextuel sélectionnez Format de la Zone automatique, allez dans l'onglet Zone de texte, et sélectionnez ce que vous souhaitez (renvoi à la ligne, ajustement).</p>
Bouton Accueil		Ce bouton préparamétré permet le retour à la première diapositive de votre diaporama
Bouton Aide		Ce bouton n'a aucune action prédéfinie, il peut vous permettre d'afficher une aide (lancement d'un UserForm par Macro par exemple).
Bouton Information		Ce bouton n'a aucune action prédéfinie, il peut vous permettre d'afficher une information (lancement d'un UserForm par Macro par exemple).
Bouton Précédent		Ce bouton préparamétré permet de revenir sur la diapositive précédente.
Bouton Suivant		Ce bouton préparamétré permet de passer à la diapositive suivante.
Bouton Début		Ce bouton préparamétré permet de revenir à la première diapositive. Il a le même effet que le bouton Accueil.
Bouton Fin		Ce bouton préparamétré permet d'aller à la dernière diapositive.
Bouton Retour		<p>Ce bouton préparamétré permet de revenir à la dernière diapositive affichée.</p> <p>Ne pas confondre avec le bouton précédent, qui revient à la diapositive qui se trouve dans l'ordre du diaporama.</p>
Bouton Document		Ce bouton n'est pas préparamétré, utilisez-le pour lancer un programme ou visualiser un autre document.
Bouton Son		<p>Ce bouton est préparamétré, pour vous permettre de lancer un son. Une liste de sons prédéfinis apparaît, mais vous pouvez toujours en jouer d'autres (allez en bas de la liste déroulante).</p>
Bouton Vidéo		Ce bouton n'est pas préparamétré, utilisez-le pour lancer une vidéo.

III-B - Affecter les boutons d'action à toutes les diapositives

Si vous avez plusieurs diapositives et que vous voulez ajouter à celles-ci les boutons "Suivant", "Précédent" et "Accueil", cela devient fastidieux de le faire pour chaque diapositive.

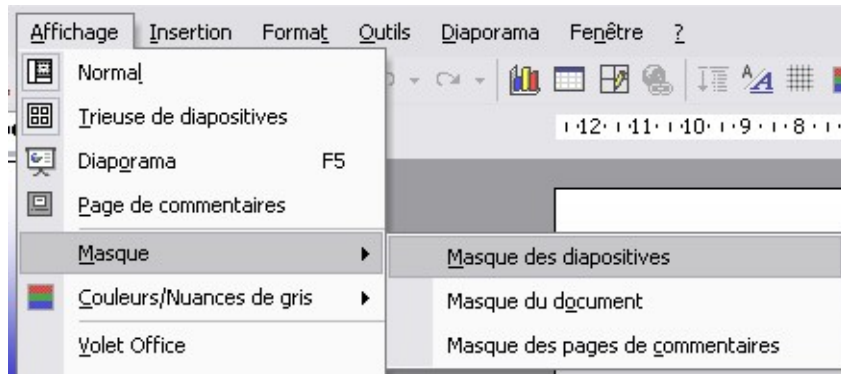
Nous allons voir différentes manières de réaliser simplement cette opération.

III-B-1 - Manuellement

Il est donc possible d'affecter manuellement nos boutons d'action à chaque diapositive.

Pour effectuer cette opération nous allons utiliser un masque de diapositive.

Allez dans le menu **Affichage/Masque/Masque** de diapositive.



Ensuite positionnez vos trois boutons d'action.

Renommez le masque PageNormal à l'aide du cinquième bouton de la barre d'outils.

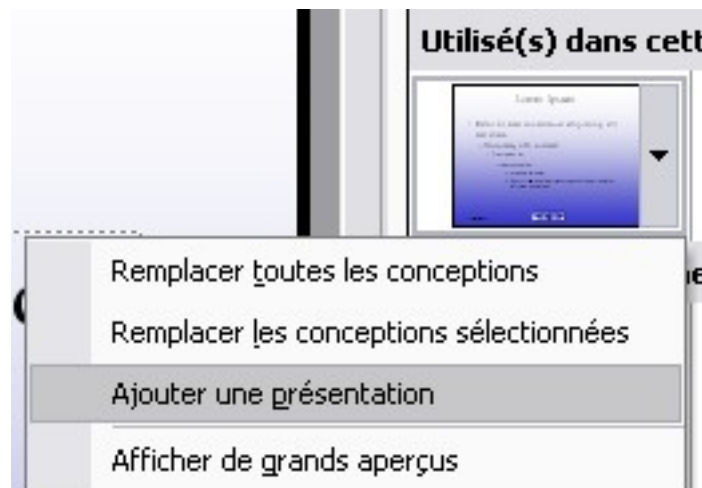
Puis cliquez sur Conserver le masque et ensuite cliquez sur Désactiver le mode Masque.



Toutes vos diapositives comporteront maintenant vos trois boutons d'action.

Cependant, l'inconvénient de cette méthode est que la première diapositive comporte les boutons Précédent et Accueil. Nous allons voir comment modifier cela.

- Repassez en mode diapositive.
- Sélectionnez la diapositive dans le volet Office.
- Sélectionnez **Ajouter une présentation**.



Vous verrez alors une deuxième présentation identique à la première s'afficher à la droite de celle-ci.

- Sélectionnez-la.
- Passez en mode masque de diapositive.
- Enlevez les boutons Précédent et Accueil.

Puis renommez-la en PremièrePage, et désactivez le mode masque.

Mais vous verrez que toutes les diapositives ont pris le nouveau masque.

Pour rétablir l'ensemble :

- Sélectionnez le premier masque et cliquez sur Appliquer à toutes les diapositives.
- Sélectionnez la première diapositive, allez dans le dernier masque (PremièrePage) et sélectionnez Appliquer aux diapositives sélectionnées.

Vous avez maintenant la première diapositive avec le bouton Suivant, et les autres avec tous les boutons.

De la même manière vous pouvez faire le masque de la dernière diapositive.

III-B-2 - Par programmation

L'ajout de boutons d'action peut également être fait grâce à du code VBA, nous allons donc créer nos boutons par programmation et les positionner.

Dans un module, on positionne en entête la déclaration des variables :

```
VBA
' ===== variables du module =====
Dim sld As Slide
Dim i As Integer
Dim shp As Shape
Dim intTopBtn As Integer ' détermine la position des boutons par rapport à la hauteur
Dim intHeightBtn As Integer ' détermine la hauteur de nos boutons
Dim intWidthBtn As Integer ' détermine la largeur de nos boutons
```

Puis toujours dans ce module on positionne les routines suivantes :

```
VBA
Public Sub BtnPrecedent()
' =====
' Procedure      : Module :mod_BoutonAction // Procédure : BtnPrecedent
' DateTime      : 07/03/2008 05:06
' Auteur        : Starec - Philippe JOCHMANS - http://starec.developpez.com
' Description    : Cette procédure crée le bouton précédent
' =====

' ===== variables =====
Dim intLeft As Integer ' position du bouton par rapport au bord gauche
intLeft = (ActivePresentation.PageSetup.SlideWidth / 2) - (intWidthBtn * 1.5)

' ajout du bouton et positionnement

Set shp = sld.Shapes.AddShape(msoShapeActionButtonBackorPrevious, intLeft, intTopBtn,
intWidthBtn, intHeightBtn)
' ation du bouton, et mise en place de la couleur
With shp
.ActionSettings(ppMouseClick).Action = ppActionPreviousSlide
.Fill.ForeColor.RGB = RGB(200, 180, 250)
.Name = "Precedent"
End With
```

VBA

```

End Sub

Public Sub BtnAccueil()
' =====
' Procedure      : Module :mod_BoutonAction // Procédure : BtnAccueil
' DateTime      : 07/03/2008 05:06
' Auteur        : Starec - Philippe JOCHMANS - http://starec.developpez.com
' Description    : Cette procédure crée le bouton précédent
' =====

' ===== variables =====
Dim intLeft As Integer ' position du bouton par rapport au bord gauche
intLeft = (ActivePresentation.PageSetup.SlideWidth / 2) - (intWidthBtn / 2)

' ajout du bouton et positionnement

Set shp = sld.Shapes.AddShape(msoShapeActionButtonHome, intLeft, intTopBtn, intWidthBtn,
intHeightBtn)
' action du bouton, et mise en place de la couleur
With shp
.ActionSettings(ppMouseClick).Action = ppActionFirstSlide
.Fill.ForeColor.RGB = RGB(200, 180, 250)
.Name = "Accueil"
End With
End Sub

Public Sub BtnSuivant()
' =====
' Procedure      : Module :mod_BoutonAction // Procédure : BtnSuivant
' DateTime      : 07/03/2008 05:06
' Auteur        : Starec - Philippe JOCHMANS - http://starec.developpez.com
' Description    : Cette procédure crée le bouton précédent
' =====

' ===== variables =====
Dim intLeft As Integer ' position du bouton par rapport au bord gauche
intLeft = (ActivePresentation.PageSetup.SlideWidth / 2) + (intWidthBtn / 2)

' ajout du bouton et positionnement

Set shp = sld.Shapes.AddShape(msoShapeActionButtonForwardorNext, intLeft, intTopBtn,
intWidthBtn, intHeightBtn)
' action du bouton, et mise en place de la couleur
With shp
.ActionSettings(ppMouseClick).Action = ppActionNextSlide
.Fill.ForeColor.RGB = RGB(200, 180, 250)
.Name = "Suivant"
End With
End Sub
    
```

Ces trois procédures ont la même structure :

- on crée un bouton avec la méthode **AddShapes**, on détermine le type de bouton d'action, on le positionne ;
- ensuite on détermine l'action du bouton, sa couleur, et son nom.

Pour lancer la création de ceux-ci dans l'ensemble du diaporama on ajoute cette procédure :

VBA

```

Public Sub AjoutBoutonAction()
' =====
' Procedure      : Module :mod_BoutonAction // Procédure : AjoutBoutonAction
' DateTime      : 05/03/2008 05:45
' Auteur        : Starec - Philippe JOCHMANS - http://starec.developpez.com
' Description    : Cette procédure va permettre d'ajouter des boutons d'action à notre
'                 diaporama
' =====
    
```

VBA

```

' ===== affectation =====
' on récupère le nombre de slides (diapositives)
i = ActivePresentation.Slides.Count
' pour positionner le haut de nos boutons
intTopBtn = ActivePresentation.PageSetup.SlideHeight - 100
' dimensions des boutons
intHeightBtn = 50
intWidthBtn = 50

For Each sld In ActivePresentation.Slides
    Select Case sld.SlideIndex
        Case 1
            ' on est sur la première diapositive
            Call BtnSuivant
        Case i
            ' on est sur la dernière diapositive
            Call BtnPrecedent
            Call BtnAccueil
        Case Else
            ' on est sur les autres diapositives
            Call BtnPrecedent
            Call BtnAccueil
            Call BtnSuivant
    End Select
Next sld
End Sub
    
```

Le principe est simple, on parcourt la liste des **slides**, et selon le cas on ajoute les boutons adéquats.

Positionnez-vous sur cette procédure et appuyez sur la touche F5, vos diapositives possèdent dorénavant des boutons de navigation.

Si vous devez ajouter une diapositive à votre diaporama, il va donc falloir supprimer tous les boutons pour les recréer par la suite.

Pour cela, positionnez cette routine dans votre module.

VBA

```

Public Sub SupBoutonAction()
' =====
' Procédure      : Module :mod_BoutonAction // Procédure : SupBoutonAction
' DateTime      : 05/03/2008 06:16
' Auteur        : Starec - Philippe JOCHMANS - http://starec.developpez.com
' Description   : Cette procédure va permettre de supprimer les boutons d'action
' =====

' ===== déclaration =====
Dim sld As Slide
Dim shp As Shape
Dim i As Integer
For i = 1 To 3
    For Each sld In ActivePresentation.Slides
        For Each shp In sld.Shapes
            If shp.Name = "Precedent" Or shp.Name = "Accueil" Or shp.Name = "Suivant" Then
                shp.Delete
            End If
        Next shp
    Next sld
Next i
End Sub
    
```

Exécutez-la, tous vos boutons seront supprimés, ensuite relancez la routine de création des boutons. Votre diaporama est maintenant à nouveau opérationnel.

i N'oubliez pas de décocher la case de parcours manuel (voir plus haut), ou d'inclure dans la création de vos boutons le code qui permet d'effectuer cette opération.

Toutes ces procédures ne sont valables que lors d'un parcours linéaire de votre diaporama.

III-C - Formatage des boutons d'action

Il n'est pas possible de modifier l'icône des boutons d'action.

Vous pouvez cependant :

- modifier la couleur comme n'importe quelle forme grâce à la barre d'outils dessin ou par l'intermédiaire du menu contextuel en sélectionnant Format de la zone automatique ;
- ajouter du texte, mais l'icône se trouvera toujours centrée, à réserver au bouton personnalisé ;
- modifier la profondeur du bouton en agissant sur le losange jaune qui apparaît lorsque vous avez sélectionné le bouton (l'icône sera redimensionnée en conséquence).

IV - LES IMAGES ET LES FORMES

À la place des boutons d'action vous pouvez créer vous-même vos propres boutons de navigation à l'aide d'images ou de formes.

Nous allons étudier le cas des formes, en sachant que pour les images le fonctionnement est similaire.

IV-A - Manuellement

Le positionnement manuel est similaire au positionnement des boutons d'action, avec l'utilisation des masques de diapositives.

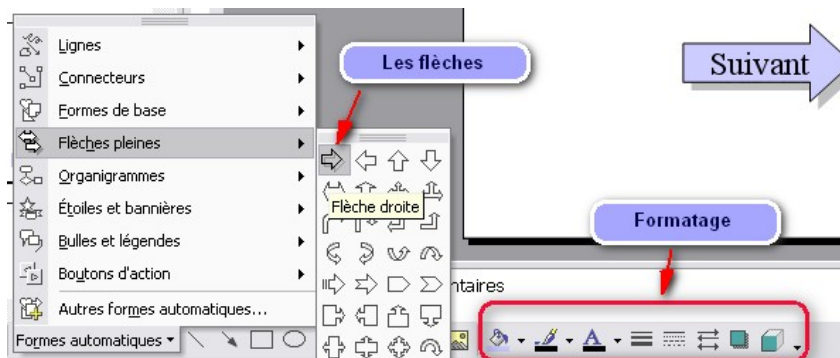
IV-A-1 - Création des formes

Nous allons juste voir comment créer nos formes par un exemple.

Création des flèches

Vous pouvez utiliser n'importe quel type de forme, mais graphiquement le plus parlant ce sont des flèches.

Vous trouverez ci-dessous la procédure pour créer une flèche :



Vous pourrez bien sûr formater celle-ci à votre goût.

Pour ajouter du texte à celle-ci, faites un clic droit sur la flèche pour visualiser le menu contextuel et sélectionnez **Ajouter Texte**.

Effectuez la même opération pour la flèche de retour, et vous pouvez utiliser une autre forme pour l'accueil.

IV-A-2 - Les actions sur les formes

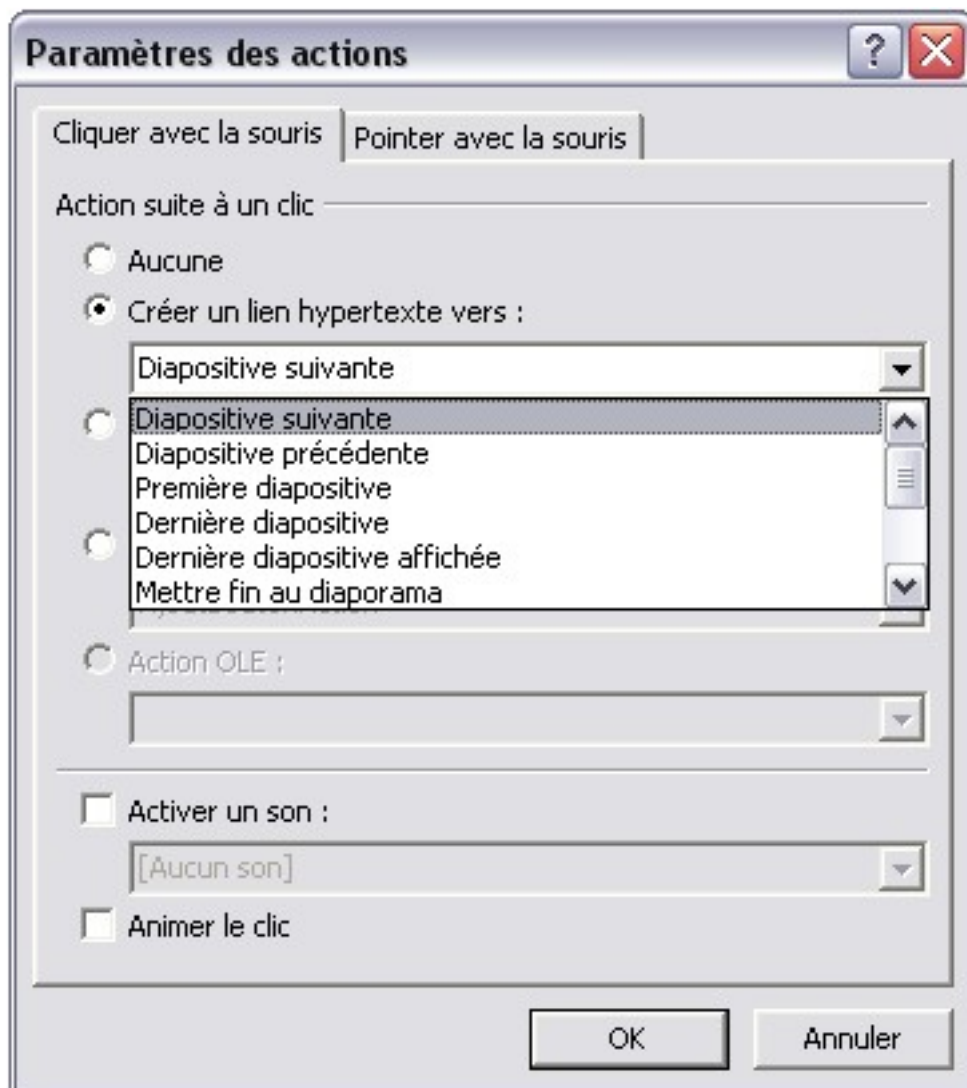
Contrairement aux boutons d'action, les formes ne sont pas préparamétrées, il vous appartient donc de réaliser cette opération.

Pour cela, faites un clic droit sur la forme pour faire apparaître le menu contextuel, et sélectionnez **Paramètres des actions**.

Vous arrivez sur l'onglet "Cliquez avec la souris", sélectionner **Créer un lien hypertexte vers**, et vous aurez une liste vous donnant plusieurs choix.

Il ne vous reste plus qu'à faire votre choix, et de cliquer sur OK en bas de la fenêtre.

Effectuez le paramétrage pour chacun de vos boutons.



En utilisant la méthode évoquée sur les masques de diapositives, vous pourrez créer un masque pour chaque type.



La manipulation pour affecter l'action doit se faire lorsque vous êtes en mode masque pour que celle-ci soit opérationnelle.

IV-B - Par programmation

Le principe sera exactement le même que pour la création des boutons d'action. La seule différence concernera la création de la forme par d'autres paramètres.

Déclaration des variables :

```
VBA
' ===== variables du module =====
Dim sld As Slide
Dim i As Integer
Dim shp As Shape
Dim intTopBtn As Integer ' détermine la position des boutons par rapport à la hauteur
Dim intHeightBtn As Integer ' détermine la hauteur de nos boutons
Dim intWidthBtn As Integer ' détermine la largeur de nos boutons
```

Ajout des Formes :

```
Public Sub AjoutFormes ()
' =====
' Procédure      : Module :mod_BoutonForm // Procédure : AjoutFormes
' DateTime       : 08/03/2008 07:44
' Auteur         : Starec - Philippe JOCHMANS - http://starec.developpez.com

' Description    : Cette routine va permettre d'ajouter les flèches "suivantes" et "précédentes"
' =====

' ===== affectation =====
' on récupère le nombre de slides (diapositives)
i = ActivePresentation.Slides.Count
' pour positionner le haut de nos boutons
intTopBtn = ActivePresentation.PageSetup.SlideHeight - 100
' dimensions des flèches
intHeightBtn = 50
intWidthBtn = 100

For Each sld In ActivePresentation.Slides
    Select Case sld.SlideIndex
        Case 1
            ' on est sur la première diapositive
            Call FlecheSuivante
        Case i
            ' on est sur la dernière diapositive
            Call FlechePrecedente
        Case Else
            ' on est sur les autres diapositives
            Call FlechePrecedente
            Call FlecheSuivante
    End Select
Next sld
End Sub
```

Création des deux flèches :

```
VBA
Public Sub FlechePrecedente ()
' =====
' Procédure      : Module :mod_BoutonForm // Procédure : FlechePrecedente
```

VBA

```
' DateTime      : 08/03/2008 07:46
' Auteur       : Starec - Philippe JOCHMANS - http://starec.developpez.com
' Description  : Cette procédure va permettre la création de la flèche précédente
' =====

' ===== variables =====
Dim intLeft As Integer ' position du bouton par rapport au bord gauche
intLeft = (ActivePresentation.PageSetup.SlideWidth / 2) - (intWidthBtn * 1.5)

' ajout du bouton et positionnement

Set shp = sld.Shapes.AddShape(msoShapeLeftArrow, intLeft, intTopBtn, intWidthBtn,
intHeightBtn)
' action du bouton, et mise en place de la couleur
With shp
    .ActionSettings(ppMouseClick).Action = ppActionPreviousSlide
    .Fill.ForeColor.RGB = RGB(200, 180, 250)
    .Name = "Précédente"
End With
' mise en place du texte
With shp.TextFrame.TextRange
    .Text = "Précédente"
    .Font.Name = "Tahoma"
    .Font.Size = 12
    .Font.Bold = msoTrue
End With
End Sub

Public Sub FlecheSuivante()
' =====
' Procédure      : Module :mod_BoutonForm // Procédure : FlecheSuivante
' DateTime      : 08/03/2008 07:46
' Auteur       : Starec - Philippe JOCHMANS - http://starec.developpez.com
' Description  : Cette procédure va permettre la création de la flèche suivante
' =====

' ===== variables =====
Dim intLeft As Integer ' position du bouton par rapport au bord gauche
intLeft = (ActivePresentation.PageSetup.SlideWidth / 2) + (intWidthBtn * 0.5)

' ajout du bouton et positionnement

Set shp = sld.Shapes.AddShape(msoShapeRightArrow, intLeft, intTopBtn, intWidthBtn,
intHeightBtn)
' action du bouton, et mise en place de la couleur
With shp
    .ActionSettings(ppMouseClick).Action = ppActionNextSlide
    .Fill.ForeColor.RGB = RGB(200, 180, 250)
    .Name = "Suivante"
End With
' mise en place du texte
With shp.TextFrame.TextRange
    .Text = "Suivante"
    .Font.Name = "Tahoma"
    .Font.Size = 12
    .Font.Bold = msoTrue
End With
End Sub
```

Suppression des flèches :

VBA

```
Public Sub SupFleche()
' =====
' Procédure      : Module :mod_BoutonForm // Procédure : SupFleche
' DateTime      : 08/03/2008 07:57
```

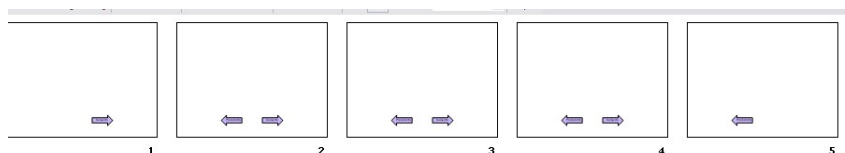
VBA

```
' Auteur      : Starec - Philippe JOCHMANS - http://starec.developpez.com
' Description  : Suppression des flèches
' =====

' ===== déclaration =====
Dim sld As Slide
Dim shp As Shape
Dim i As Integer
For i = 1 To 2
    For Each sld In ActivePresentation.Slides
        For Each shp In sld.Shapes
            If shp.Name = "Précédente" Or shp.Name = "Suivante" Then
                shp.Delete
            End If
        Next shp
    Next sld
Next i
End Sub
```

Comme vous le voyez, le fonctionnement est similaire, les différences interviennent dans le type de Shape que l'on rajoute, et le texte.

Vous aurez alors cela :



*N'hésitez pas à consulter l'explorateur d'objet (F2) et l'aide (F1), vous y trouverez tous les paramètres.
Ces deux touches doivent être les plus usées de votre clavier.*

IV-C - Cas particulier des images

Le principe avec des images est exactement le même, sauf que lorsque vous voulez ajouter une image, il vous faut utiliser la méthode **AddPicture** de la collection **Shapes**.

V - LES LIENS HYPERTEXTES

Il est possible de naviguer grâce à des liens hypertextes (autres que ceux sur les boutons d'action).

Les liens hypertextes vont vous permettre d'effectuer une action non linéaire, comme on le voit sur le premier schéma de cet article.

Vous en trouverez un exemple et la manière de procéder dans cet article [Les listes à puces et numérotées dans PowerPoint](#).

Les liens hypertextes sur la deuxième diapositive (la première servant en général de page d'accueil), couplés avec les méthodes ci-dessus affectées à un bloc de diapositives permettent de réaliser des transitions non linéaires facilement.

VI - CONCLUSION

En conclusion, lorsque vous voulez gérer vous-même la navigation dans vos diaporamas, il faut rester logique.

- **Navigation linéaire** : choisissez les boutons d'action qui sont prédéfinis ou les formes.
- **Navigation non linéaire** : vous pouvez réaliser une application non linéaire grâce aux liens hypertextes qui vous permettront de naviguer comme dans une table des matières d'un document Word.

Pensez que les personnes qui utiliseront vos diaporamas ne sont peut-être pas habituées à les manipuler. Le fait de mettre des indications comme des flèches ou autres facilitera grandement l'utilisation de ceux-ci, de plus chaque lecteur ne lit pas de la même façon et n'a pas la même vitesse de lecture.

VII - REMERCIEMENTS

Je tiens à remercier **Dolphy35** , **Jeannot45** et **ClaudeLELOUP** pour leur relecture et leurs conseils judicieux. Toute l'équipe de la rédaction et les membres du forum qui nous aident à nous surpasser.