

## Les bases du contrôle MSChart avec Visual Basic 6.0

### Sommaire

- I. Création d'un graphique 2D
  - a. Base du graphique 2 D
  - b. Titres, légende et note (footnote)
  - c. Couleurs de trait et du graphe
  - d. Copier/coller le graphe
  - e. Deuxième série de points
- II. Création d'un graphique 3D
  - a. Base du graphique 3D
  - b. Encore quelques instructions de graphismes
  - c. Orientation d'un graphique
  - d. Modification des valeurs de départ – Affichage sur le graphe
  - e. Trucs et astuces
  - f. Impression
- III. Téléchargements

Le contrôle MSChart (MSCHART32.OCX) permet de réaliser des graphiques de différents types. Je vous propose de passer en revue quelques unes des possibilités offertes par ce contrôle en commençant par un graphique très simple puis en « l'habillant » avec les propriétés, méthodes et événements supportés par le MSChart.

Nous utiliserons aussi la liaison avec une source de données Access.

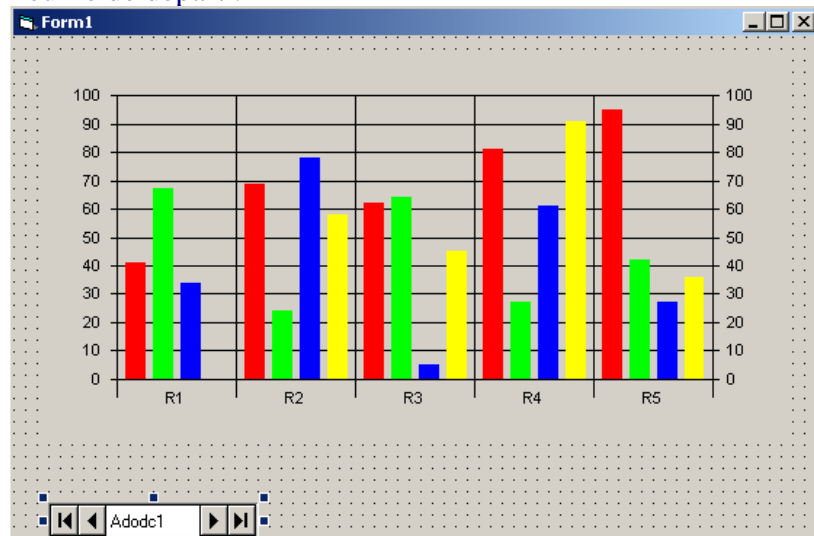
### I. Création d'un graphique simple 2D

Sur une feuille, placez un contrôle MSChart. Le nom de ce dernier restera celui généré par VB : MSChart1  
La source de données est une simple table Access liée par un contrôle Adodc.(adodc1 sur la feuille). Il n'y a aucun contrôle des enregistrements, ceci n'étant pas le but de ce cours.

Composition de la table Access :

Climatologie : Table			
	Mois	TempMax	TempMin
	Janvier	4,3	-1,2
	Février	6,7	0,3
	Mars	10,3	2,2
	Avril	14,2	5,1
	Mai	18,4	7,9
	Juin	22	10,9
	Juillet	22,7	12,1
	Août	22,3	12,2
	Septembre	20,5	10,6
	Octobre	15,4	7,3
	Novembre	8,9	3,1
	Décembre	5,6	0,2

Feuille de départ :



NB : le contrôle adodc1 peut être mis en Visible = False pour une visualisation optimisée <sup>(1)</sup>

### a) Base du graphique 2D

Private Sub Form\_Load()

```
' Initialisation de l'accès aux données
Dim MaBD As String
Dim i As Integer

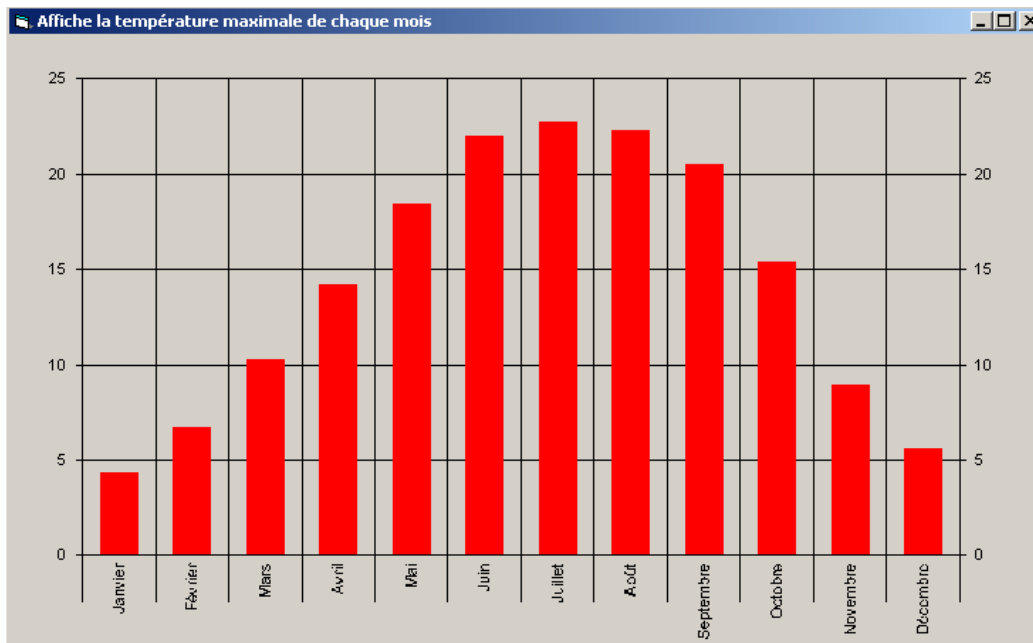
MaBD = App.Path & "\Chart.mdb"
With Adodc1
    .ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & MaBD & ";Persist Security
Info=False"
    .RecordSource = "SELECT * FROM Climatologie"
End With
Adodc1.Refresh
Adodc1.Recordset.MoveFirst

' Initialisation du graphique simple
MSChart1.AllowSelections = False ' Interdit la sélection dans le graphe
With MSChart1
    .chartType = VtChChartType2dBar ' type de graphe
    .ColumnCount = 1 ' nombre de barre
    .RowCount = 12 ' nombre de mois
    ' lecture du nom des mois et des températures dans la BD
    For i = 1 To 12
        .Row = i
        .Column = 1
        .RowLabel = Adodc1.Recordset.Fields(0) ' le nom de chaque mois
        .Data = Adodc1.Recordset.Fields(1).Value ' la valeur de température maximale
        MoyMax = MoyMax + Adodc1.Recordset.Fields(1) ' cumul des valeurs pour calculer la moyenne
        Adodc1.Recordset.MoveNext ' le mois suivant
    Next i
End With
MoyMax = Format(MoyMax / 12, "#0.0")
```

```

' Pour que le curseur soit différent lorsque la souris passe sur le graphe
MSChart1.DoSetCursor = True
MSChart1.MousePointer = VtMousePointerArrowQuestion
End Sub

```



Avant de poursuivre l'exploration de la présentation graphique, il y a quelques petites notions importantes à signaler :

**PLOT** : fait référence à la zone sur laquelle le graphe est affiché.

**SERIESCOLLECTION** : permet d'atteindre via l'indice une série précise du graphe

**SERIES (objet)** : permet d'accéder aux éléments d'une collection **SeriesCollection** représentant un groupe de points de données sur un graphique.

Les références à ces propriétés sont permanentes lors de la création d'un graphe.

## b) Titres, légende et notes.

### LEGENDE

```
MSChart1.ShowLegend = True
```

```
' Texte de la légende
```

```
With MSChart1.Plot.SeriesCollection(1)
```

```
.LegendText = "T° max"
```

```
End With
```

```
With MSChart1.Legend
```

```
.Location.LocationType = VtChLocationTypeTop
```

```
.TextLayout.HorzAlignment = VtHorizontalAlignmentCenter
```

```
.VtFont.VtColor.Set 255, 255, 0
```

```
.Backdrop.Fill.Style = VtFillStyleBrush
```

```
.Backdrop.Fill.Brush.Style = VtBrushStyleHatched
```

```
.Backdrop.Fill.Brush.FillColor.Set 159, 0, 255
```

```
' La légende sera au-dessus
```

```
' Alignement centré
```

```
' Texte jaune.
```

```
' Style de fond
```

```
' Style de remplissage
```

```
' Couleur de remplissage du fond de la légende
```

```
End With
```

## TITRES

' Titre du graphe et mise en forme du texte

```
MSChart1.TitleText = "Températures annuelles"
```

```
With MSChart1.Title.VtFont
```

```
.Name = "Comic Sans MS"
```

```
.Size = 15
```

```
.Effect = VtFontEffectUnderline
```

```
End With
```

' Titre pour l'axe des Y côté gauche

```
With MSChart1.Plot.Axis(1, 1)
```

```
.AxisTitle.VtFont.Size = 14
```

```
.AxisTitle.Visible = True
```

```
.AxisTitle.Text = "Températures"
```

```
End With
```

' Titre pour l'axe des Y côté droit

```
With MSChart1.Plot.Axis(2, 1)
```

```
.AxisTitle.VtFont.Size = 14
```

```
.AxisTitle.Visible = True
```

```
.AxisTitle.Text = "Températures"
```

```
End With
```

' Titre pour l'axe des X

```
With MSChart1.Plot.Axis(0, 1)
```

```
.AxisTitle.VtFont.Size = 14
```

```
.AxisTitle.Visible = True
```

```
.AxisTitle.Text = "Mois"
```

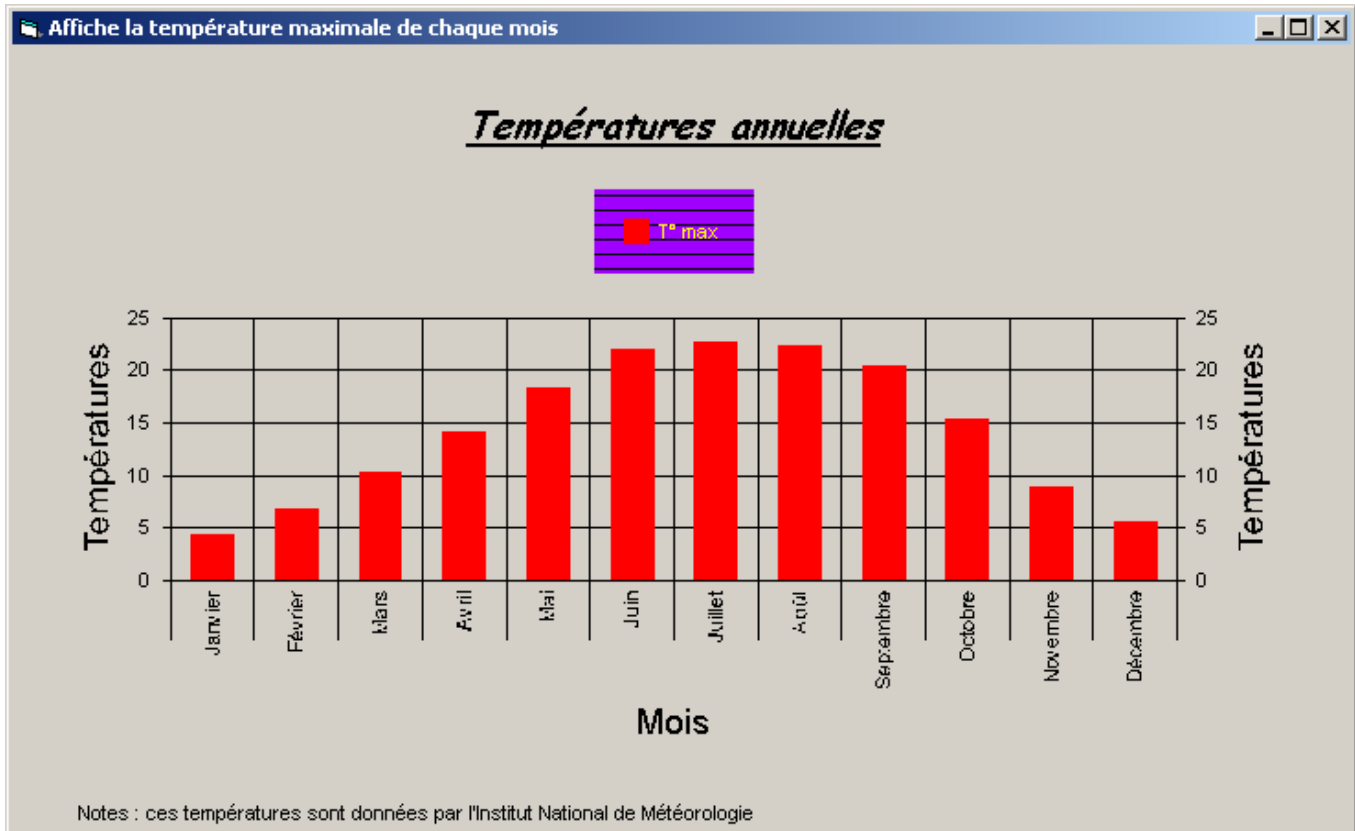
```
End With
```

## NOTE

' Note de bas de graphique (footnote)

```
MSChart1.FooterText = "Notes : ces températures sont données par l'Institut National de Météorologie"
```

Vous devez obtenir un affichage de ce type :



### c) Couleurs de trait et du graphe

' Changer la couleur du graphe

```
With MSChart1.Plot.SeriesCollection(1)  
  .DataPoints(-1).Brush.FillColor.Set 45, 44, 78 ' couleur RGB  
End With
```

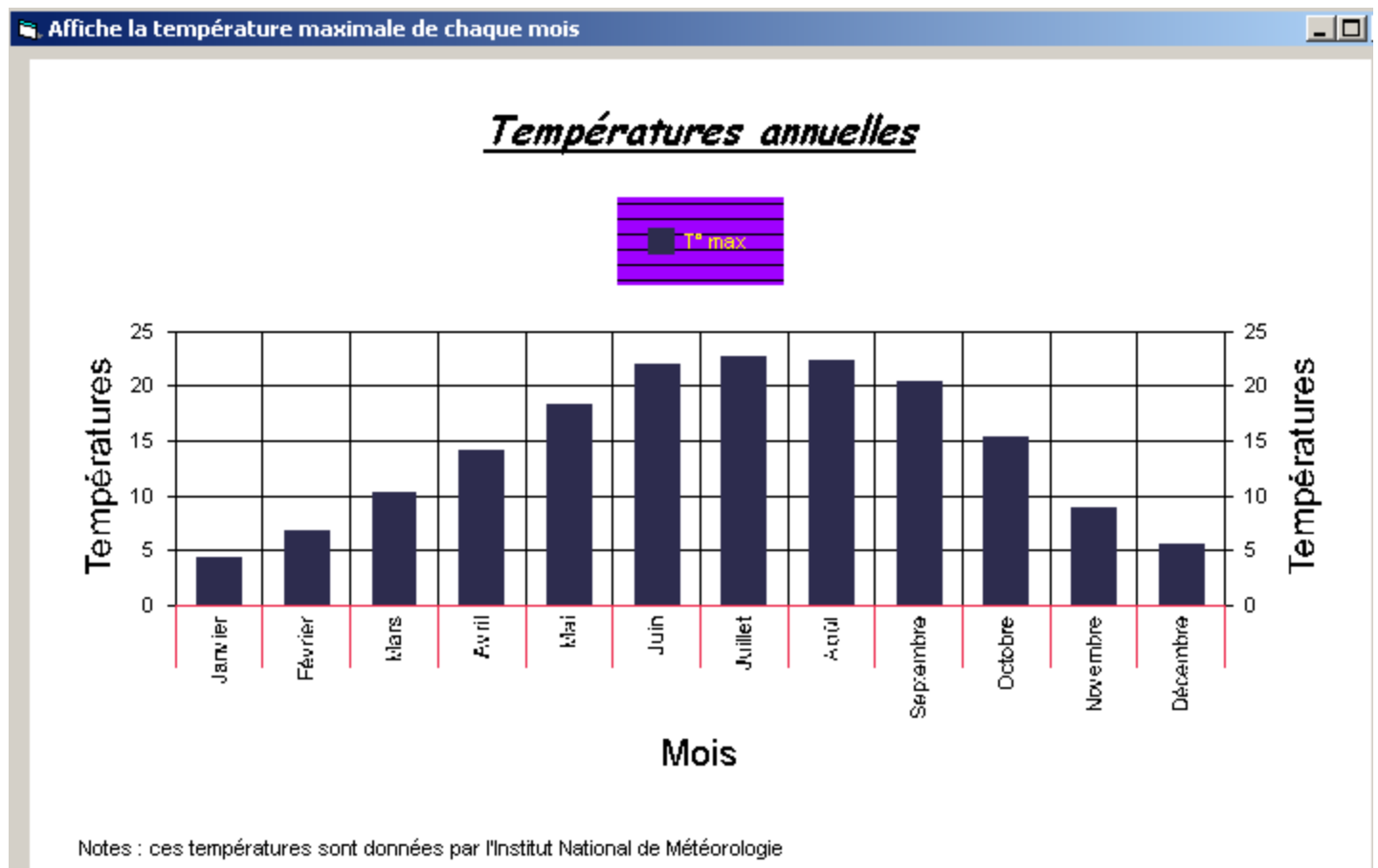
' Changer la couleur du fond du graphe

```
With MSChart1.Backdrop.Fill  
  .Style = VtFillStyleBrush  
  .Brush.FillColor.Set 255, 255, 255 'fond blanc  
End With
```

' Placer une ligne colorée entre les éléments de la légende de l'axe Y ainsi que la ligne 0

```
With MSChart1.Plot.Axis(VtChAxisIdX, 1)  
  .Pen.VtColor.Set 237, 44, 78  
End With
```

Exemple du graphe après ces modifications :



#### d) Copier/Coller le graphe

Si vous désirez afficher une image de votre graphe dans une ImageBox, par exemple, placez un contrôle image (Image1) sur une feuille ainsi que le code suivant :

```
Image1.Stretch = True  
MSChart1.EditCopy  
Image1.Picture = Clipboard.GetData(vbCFBitmap)
```

#### e) Deuxième série de points

Sur le même principe que la première série, ajoutons les températures annuelles minimales, calculons et affichons les moyennes annuelles.

```
MsgBox "Nombre axe : " & Str(MSChart1.ActiveSeriesCount) ' Nombre de séries actuellement  
With MSChart1  
.ColumnCount = 2 ' nombre de séries  
.RowCount = 12 ' nombre de mois  
Adodc1.Recordset.MoveFirst  
For i = 1 To 12  
.Row = i  
.Column = 2  
.Data = Adodc1.Recordset.Fields(2).Value ' la valeur de température minimale
```

```

MoyMin = MoyMin + Adodc1.Recordset.Fields(2)
Adodc1.Recordset.MoveNext
Next i
.Plot.SeriesCollection(2).LegendText = "T° min"
End With
MsgBox "Nombre axe : " & Str(MSChart1.ActiveSeriesCount) ' doit valoir 2 maintenant
MoyMin = Format(MoyMin / 12, "#0.0")

```

Une moyenne n'ayant pas plusieurs valeurs, elle sera représentée par une ligne sur notre graphe. Dès lors, un problème se pose car si nous ajoutons une série de données avec la valeur de cette moyenne, le résultat obtenu sera complètement erroné. Nous aurons 3 barres dont une ayant la même valeur pour tous les mois de l'année. Afin d'arriver au résultat escompté, il faut redéfinir le type de graphe à utiliser.

' Changement du type de graphe pour afficher les valeurs linéaires des moyennes

```

With MSChart1
.chartType = VtChChartType2dCombination
.ColumnCount = 4
For i = 1 To 12
.Row = i
.Column = 3
.Data = MoyMax
.Column = 4
.Data = MoyMin
Next i
.Plot.SeriesCollection(1).SeriesType = VtChSeriesType2dBar
.Plot.SeriesCollection(2).SeriesType = VtChSeriesType2dBar
.Plot.SeriesCollection(3).SeriesType = VtChSeriesType2dLine
.Plot.SeriesCollection(4).SeriesType = VtChSeriesType2dLine
.Plot.SeriesCollection(3).LegendText = "Moy max"
.Plot.SeriesCollection(4).LegendText = "Moy min"
.Refresh
End With

```

Une autre amélioration possible est l'affichage des valeurs pour chaque série lorsque la souris est positionnée sur cette série. Pour cela, on utilise l'événement `MouseMove` du graphe.

```

Private Sub MSChart1_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
Dim Partie As Integer, Index1 As Integer, Index2 As Integer, Index3 As Integer, Index4 As Integer

```

```

' Annule sélection de la courbe précédente
MSChart1.ToolTipText = ""
MSChart1.SelectPart 4, 0, 0, 0 ' 4 est une valeur de VtChChartType, voir tableau de correspondance
dans MSDN

```

```

' Calcul l'emplacement actuel
MSChart1.TwipsToChartPart X, Y, Partie, Index1, Index2, Index3, Index4
' Parcourt les courbes pour trouver sur quelle série se trouve le curseur
For i = 1 To 4
If Partie = 5 And Index1 = i Then ' on est sur une série
MSChart1.SelectPart 7, i, 0, 0 ' sélectionne la courbe sur laquelle le curseur se trouve
MSChart1.TwipsToChartPart X, Y, Partie, Index1, Index2, Index3, Index4 ' sélectionne les valeurs
If Index1 <> 0 And Index2 <> 0 Then

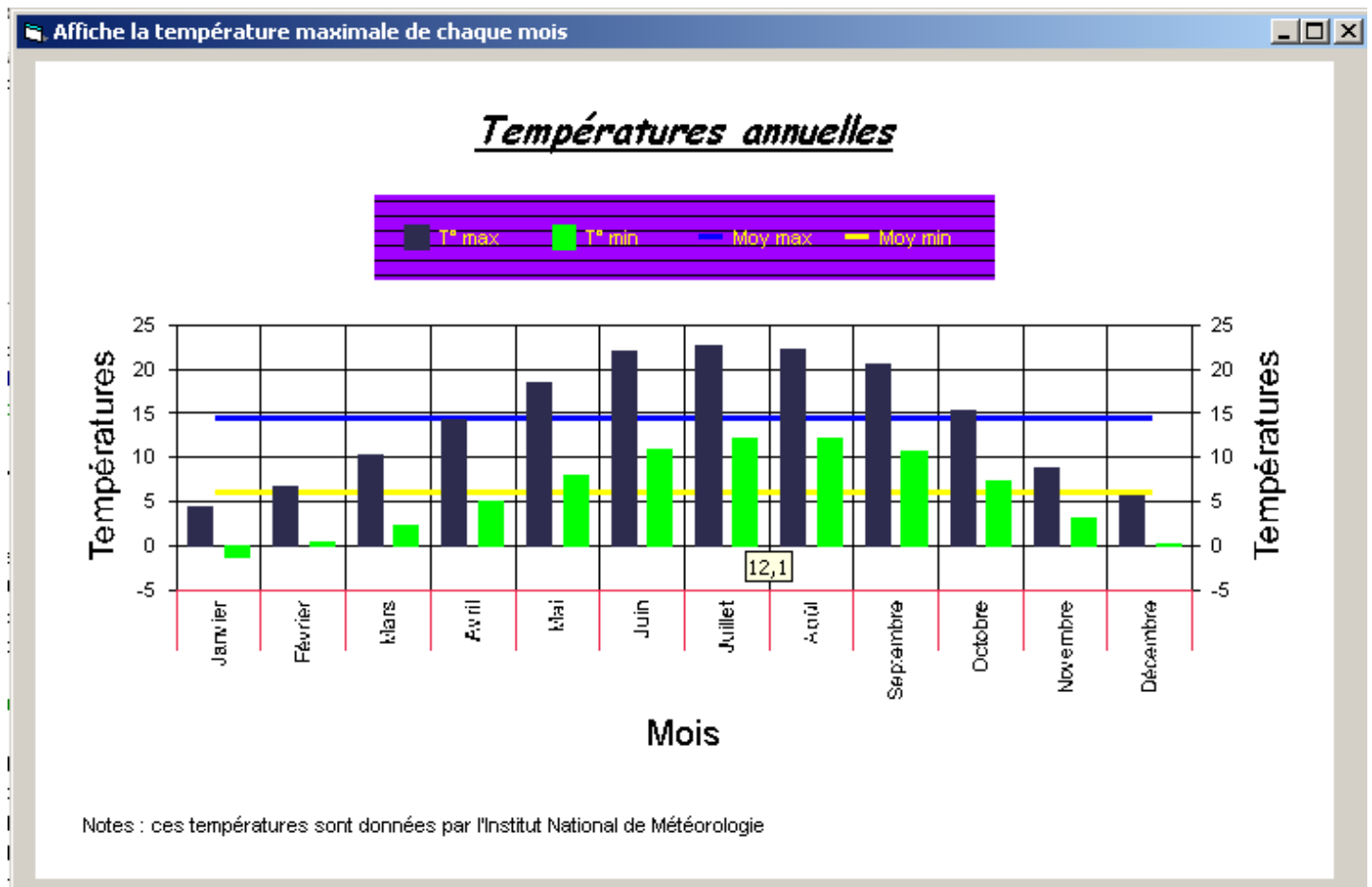
```

```

MSChart1.Column = Index1      ' valeur de l'abscisse
MSChart1.Row = Index2         ' valeur de l'ordonnée
MSChart1.ToolTipText = MSChart1.Data & " C°" ' affiche la valeur dans l'infobulle
End If
End If
Next i
End Sub

```

Après toutes ces modifications, votre graphique a l'aspect suivant :



Nous voici arrivés au terme de notre périple dans les graphes en 2 dimensions. Il y a encore certainement d'autres possibilités que vous découvrirez en « jouant » avec les propriétés du MSChart. Il vous sera très facile de vous orienter dès que la notion de Plot, SeriesCollection et Serie sera acquise et j'espère que ces quelques exemples vous auront aidé.

Abordons maintenant la 3<sup>ème</sup> dimension !



## II. Création d'un graphique 3D

Le principe est strictement le même que pour les graphiques 2D, je ne reviendrai donc plus sur les titres de graphes, légendes et autres modifications de couleurs.

La source de données est une simple table Access liée directement au graphe. Pour changer, nous allons tracer la courbe du volume des intempéries pour chaque mois.

Composition de la table Access :

Intempéries : Table		
	Mois	Humidite
▶	Janvier	83
	Février	67
	Mars	47
	Avril	53
	Mai	42
	Juin	55
	Juillet	97
	Août	83
	Septembre	69
	Octobre	90
	Novembre	90
	Décembre	67
*		0

Cocher les références ADO dans votre projet. <sup>(1)</sup>

### a) Base du graphique 3D

Private Sub Form\_Load()

‘ Initialisation de l'accès aux données

Dim MaBD As String

Dim Cnn As ADODB.Connection

Dim Rs As ADODB.Recordset

Dim CnnStr As String

Set Cnn = New ADODB.Connection

Set Rs = New ADODB.Recordset

MaBD = App.Path & "\\Chart.mdb"

CnnStr = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & MaBD & ";Persist Security Info=False"

Cnn.Open CnnStr

Rs.Open "SELECT \* FROM Intempéries", Cnn, adOpenKeyset, adLockReadOnly

‘ Alloue les données au graphique

Set MSChart1.DataSource = Rs

MSChart1.chartType = VtChChartType3dLine

End Sub

## b) Encore quelques instructions de graphismes

Une petite variante pour afficher le titre du graphe avec un effet d'ombrage.

```
With MSChart1.Title
.Location.Visible = True
.Text = "Intempéries mensuelles"
.Backdrop.Frame.Width = 1
.Backdrop.Frame.FrameColor.Set 45, 255, 125

.Backdrop.Frame.Style = VtFrameStyleDoubleLine
.Backdrop.Shadow.Style = VtShadowStyleDrop

.Backdrop.Shadow.Offset.X = 10
.Backdrop.Shadow.Offset.Y = 10
End With
```

Un fond de couleur avec un cadre pour mettre le graphique en valeur et vous pouvez toujours mettre une légende aux axes.

### FOND DU GRAPHE

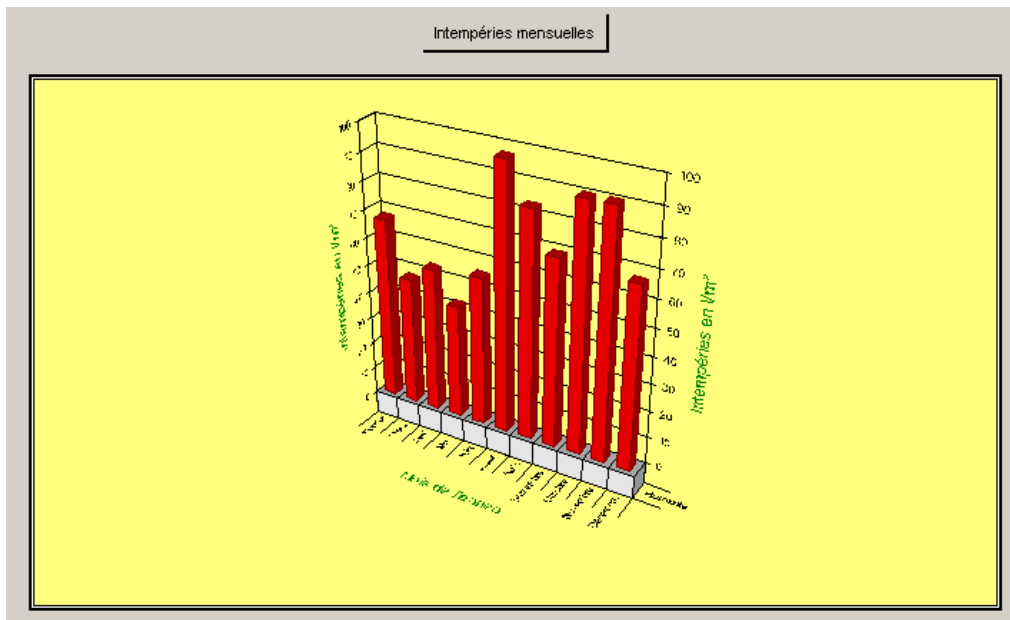
```
With MSChart1.Plot.Backdrop
.Frame.Style = VtFrameStyleThickOuter
.Fill.Style = VtFillStyleBrush
.Fill.Brush.FillColor.Set 255, 255, 125
End With
```

### TITRES DES AXES

```
With MSChart1.Plot.Axis(VtChAxisIdX).AxisTitle
.VtFont.Name = "arial"
.VtFont.Size = 12
.VtFont.VtColor.Set 0, 127, 0
.Text = "Mois de l'année "
.Visible = True
End With
```

Il suffit de remplacer "VtChAxisIdX" par "VtChAxisIdY" et/ou "VtChAxisIdY2" pour mettre un titre à l'axe Y et Y2 respectivement.

Le graphique obtenu est le suivant :



Voici encore quelques instructions qui modifient l'apparence du graphe que vous pourrez explorer vous-même car elles ne sont pas utilisées dans notre exemple.

## LUMINOSITE

Pourcentage de lumière ambiante

`MSChart1.Plot.Light.AmbientIntensity = 1` ' donne 100% d'intensité

Intensité de la lumière utilisée pour dessiner les bords des objets

`MSChart1.Plot.Light.EdgeIntensity = 0.5` ' donne 50% d'intensité

Représente la source lumineuse utilisée pour éclairer les éléments puis l'intensité de cette source

`MSChart1.Plot.Light.LightSources(0).Intensity = 1`

## BASE GRAPHIQUE ET RATIO

Pour modifier l'aspect ainsi que la taille de la base du graphe.

With `MSChart1.Plot.PlotBase`

' Modifie la hauteur de la base.

`.BaseHeight = 20`

' Utilise un style de motif pour la base.

`.Brush.Style = VtBrushStylePattern`

`.Brush.Index = VtBrushPatternHorizontal`

`.Brush.FillColor.Set 255, 160, 160`

`.Brush.PatternColor.Set 180, 180, 255`

`.Pen.Style = VtPenStyleSolid`

`.Pen.VtColor.Set 72, 72, 255`

End With

Pourcentage de la hauteur du graphique à utiliser comme profondeur.

`MSChart1.Plot.DepthToHeightRatio = 2.5`

Pourcentage de la hauteur du graphique à utiliser comme largeur.

`MSChart1.Plot.WidthToHeightRatio = 2.5`

### c) Orientation d'un graphique

Comme tout objet 3D, le graphique peut être orienté c'est-à-dire que l'on peut le faire pivoter d'un certain angle pour choisir la vision la plus appropriée. Pour visualiser les différentes manières d'afficher un graphe 3D, vous pouvez vous référer à la source jointe « Graphe3D ». Voici les exemples de syntaxe :

#### CHANGER LA PROJECTION

`MSChart1.Plot.Projection = VtProjectionTypeOblique`

#### ELEVATION ET ROTATION

`MSChart1.Plot.View3d.Elevation = 0`

L'élévation est toute valeur de 0 à 90 degrés, unité par défaut excepté si vous avez changé cette unité par une instruction du type :

`MSChart1.Plot.AngleUnit = VtAngleUnitsGrads` ou `VtAngleUnitsRadians`

`MSChart1.Plot.View3d.Rotation = 90`

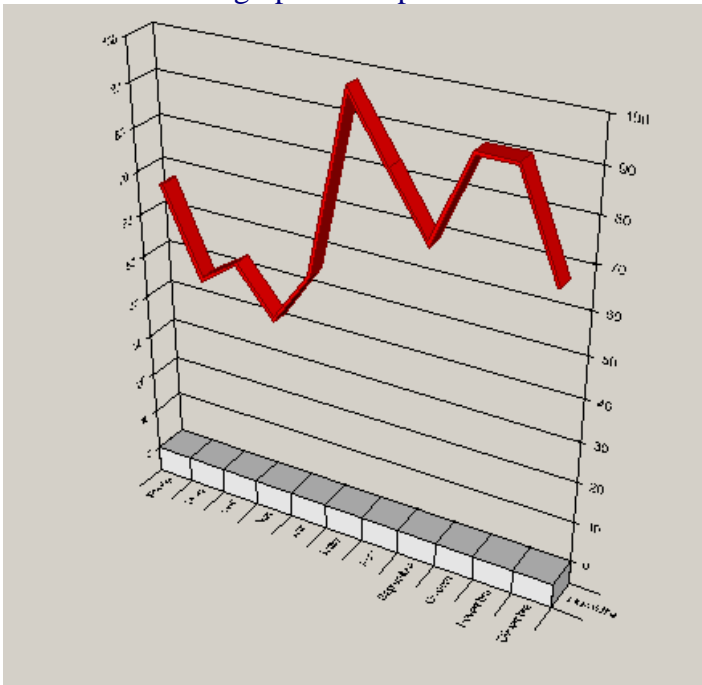
La rotation est toute valeur de 0 à 360 degrés. La remarque au niveau des unités est aussi valable que pour l'élévation

On peut combiner l'élévation et la rotation en une seule instruction :

`MSChart1.Plot.View3d.Set 90, 10` (Rotation, Elevation)

### d) Modification des valeurs – Affichage sur le graphe

Revenons à notre graphe de départ mais choisissons le type « Ligne ». Ce qui donne :



Par défaut, l'axe des Y commence à 0 mais, il y a possibilité de modifier cette valeur de départ. Dans le cas actuel, le graphe pourrait commencer à 40 pour se terminer à 95.

```
With MSChart1.Plot.Axis(VtChAxisIdY).ValueScale
    .Auto = False
    .Maximum = 95
    .Minimum = 40
End With
```

Il est aussi possible d'afficher les valeurs des points du graphe. Ce sont les labels de valeurs (DataPointLabel).

```
With MSChart1.Plot.SeriesCollection(1).DataPoints(-1)
    .Marker.Visible = True
    With .DataPointLabel
        .Custom = False
        .Component = VtChLabelComponentValue
        .LocationType = VtChLabelLocationTypeAbovePoint
    End With
End With
```

Le graphe est aussi très grand et vous désirez diminuer ses proportions

```
With MSChart1.Plot
    .AutoLayout = False
    .LocationRect.Min.X = .LocationRect.Min.X * 0.8
    .LocationRect.Min.Y = .LocationRect.Min.Y * 0.8
    .LocationRect.Max.X = .LocationRect.Max.X * 0.8
    .LocationRect.Max.Y = .LocationRect.Max.Y * 0.8
End With
```

On peut aussi changer l'aspect de la ligne

```
With MSChart1.Plot.SeriesCollection(1)
    .ShowGuideLine(VtChAxisIdY) = True
    .GuideLinePen.Style = VtPenStyleDitted
    .Pen.Style = VtPenStyleDitted
End With
```

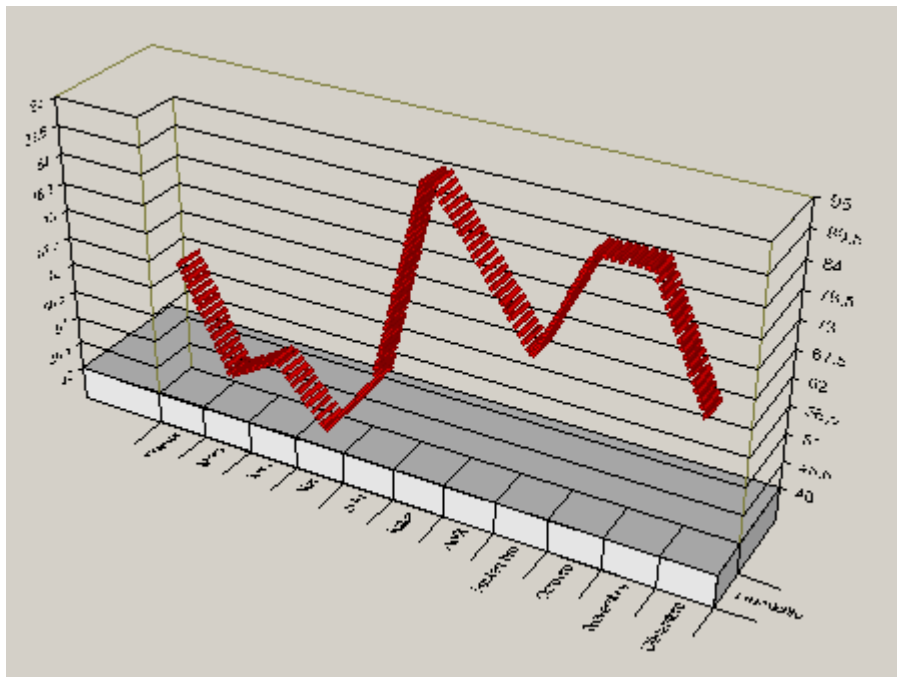
Ou encore modifier l'espacement entre les divisions sur les axes pour aérer le graphe

```
With MSChart1.Plot
    .DepthToHeightRatio = 2
    .WidthToHeightRatio = 2
    .xGap = 0
    .zGap = 0.8
End With
```

Et pour terminer, encore un peu de graphisme, dessiner un « mur » en support du graphe

```
MSChart1.Plot.Wall.Width = 1000 ' détermine « l'épaisseur » du support  
MSChart1.Plot.Wall.Pen.VtColor.Set 255, 255, 125
```

Ce qui nous amène à un graphique comme suit :



**Les valeurs de modification sont évidemment exagérées afin de bien visualiser les changements.**

### e) Trucs et Astuces

Si vous avez plusieurs séries de données, il ne faut pas mettre un code par série mais vous pouvez utiliser l'instruction « For Each » :

```
Dim Ser As Series
```

```
For Each Ser In MSChart1.Plot.SeriesCollection
```

```
    Ser. ....
```

```
Next
```

Les marqueurs ne vous plaisent pas ? il suffit de les changer comme suit :

```
With MSChart1.Plot.SeriesCollection(1)
```

```
    .SeriesMarker.Auto = False
```

' pour passer en mode manuel de choix de marqueur

```
    With .DataPoints(-1).Marker
```

```
        .Visible = True
```

```
        .Style = VtMarkerStyleDownTriangle
```

' à spécifier pour choisir la forme de marqueur

```
        .Size = 30
```

' à spécifier pour choisir la taille du marqueur

```
        .Pen.VtColor.Set 255, 255, 125
```

```
    End With
```

```
End With
```

## f) Impression

Voici une solution pour imprimer votre graphique :

Dim Titre As String

Titre = "Ceci est un exemple d'impression d'un graphe"

Mchart1.EditCopy

With Printer

.CurrentX = (.ScaleWidth - .TextWidth(Titre)) / 2

Printer.Print Titre

.PaintPicture Clipboard.GetData, 0, .CurrentY, Printer.ScaleWidth

.EndDoc

End With

Printer.ScaleWidth imprime le graphe sur toute la largeur de la feuille. Il suffit de placer une valeur pour adapter la grandeur du graphe selon vos besoins. Attention, cette valeur dépend du ScaleMode dans lequel vous avez placé l'imprimante.

Nous voici arrivés au terme de notre survol du contrôle MSChart. Comme vous pouvez le constater, bien des découvertes sont encore à faire. Bon travail !

## III. Téléchargements

Le code source des graphiques 2D et 3D sont accessibles via la page d'accueil de ce tutoriel.

*Les sources présentés sur cette pages sont libres de droits, et vous pouvez les utiliser à votre convenance. Par contre la page de présentation de ces sources constitue une oeuvre intellectuelle protégée par les droits d'auteurs. Copyright © 2004 Cécile MUNO (Khany) . Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à 3 ans de prison et jusqu'à 300 000 E de dommages et intérêts.*

---

<sup>(1)</sup> Pour les références ADO, parcourez les nombreux cours accessibles sur la page des cours du forum Visual Basic du site [www.developpez.com](http://www.developpez.com).

This document was created with Win2PDF available at <http://www.daneprairie.com>.  
The unregistered version of Win2PDF is for evaluation or non-commercial use only.