Sumari

SUMARI _____ **1**

I. LLISTATS DE CLASSES _____ **3**

A. Classe CAcció 3

B. Classe CBarra 3

C. Classe Cnode 28

II. LLISTAT DE MÒDULS _____ **34**

A. Accés a la base de dades 34

B. Representació gràfica 76

I. Llistats de classes

A. Classe CAcció

```
Option Explicit

Private m_NomAccio As String
Private m_NumeroAccio As Long
Private m_ColorRGBAccio(1 To 3) As Integer

Property Get NomAccio() As String
NomAccio = m_NomAccio
End Property
Property Let NomAccio(ByVal Val As String)
m_NomAccio = Val
End Property

Property Get NumeroAccio() As Long
NumeroAccio = m_NumeroAccio
End Property
Property Let NumeroAccio(ByVal Val As Long)
m_NumeroAccio = Val
End Property

Property Get ColorRGBAccio(indexColor As Integer) As Integer
ColorRGBAccio = m_ColorRGBAccio(indexColor)
End Property
Property Let ColorRGBAccio(indexColor As Integer, ByVal Val As Integer)
m_ColorRGBAccio(indexColor) = Val
End Property
```

B. Classe CBarra

```
Option Explicit
Private m_X1 As Double
Private m_Y1 As Double
Private m_Z1 As Double
Private m_X2 As Double
Private m_Y2 As Double
Private m_Z2 As Double
Private m_Numero As Long
Private m_NumNode1 As Long
Private m_NumNode2 As Long
Private m_E As Double
Private m_Ix As Double
Private m_Iy As Double
Private m_GJ As Double
```

```
Private m_A As Double
Private m_Nom As String
Private m_Denominacio As String
Private m_L As Double
Private m_VxOrto As Double
Private m_VyOrto As Double
Private m_VzOrto As Double
Private m_AngleOrientacio As Double
Private m_FxNode1 As Double
Private m_FyNode1 As Double
Private m_FzNode1 As Double
Private m_MxNode1 As Double
Private m_MyNode1 As Double
Private m_MzNode1 As Double
Private m_FxNode2 As Double
Private m_FyNode2 As Double
Private m_FzNode2 As Double
Private m_MxNode2 As Double
Private m_MyNode2 As Double
Private m_MzNode2 As Double
Private m_MatriuDeRigidesaLocal As Variant
Private m_MatriuDeRigidesaGlobalElemental As Variant
Private m_MatriuDeRigidesaGlobalCalculat As Boolean
Private m_MatriuRotacional As Variant
Private m_VectorCorriments As Variant
Private m_ColorTextBarra As Double
Private m_Seleccionat As Boolean
Private m_Node1Articulat As Boolean
Private m_Node2Articulat As Boolean
Private m_TipusDeCarregaRepartida As Integer
'1 Cap Força repartida
'2 Pressió segons els eixos locals
'3 Pressió segons els eixos globals
Private m_CarregaIni As Double
Private m_CarregaFi As Double
Private m_ZCarregaIni As Double
Private m_ZCarregaFi As Double
Private m_AngleAlfaCREixosGlobals As Double
Private m_AngleBetaCREixosGlobals As Double
Private m_AngleAlfaCREixosLocals As Double


Private m_TipusPerfil As Integer
'1 Rectangular
'2 Circular
'3 IPN, IPE, HEB, HEA, HEM
'4 UPN
'5 L
'6 HUECO REDONDO
'7 HUECO CUADRADO
'8 HUECO RECTANGULAR
```

```
Private m_B As Double
Private m_H As Double
Private m_R As Double
Private m_DE As Double
Private m_DE1 As Double
Private m_Material As String
Private m_Densitat As Double
Private m_CoefDilatacioTermica As Double
Private m_CaresVisibles As Boolean
Private m_OrdreDeDibuix As Long
Private m_OrdreDeDibuixSeleccionat As Boolean
Private m_ProfunditatZ As Double
Private m_Ip As Double
Private m_G As Double
Private m_Nu As Double
Private m_RMaterial As Integer
Private m_GMaterial As Integer
Private m_BMaterial As Integer
Private m_UnitatsPreferencials As String
Private m_ColCarreguesRepartides As CColCarregaRepartida
Private m_ColForcesPuntualsAgrupadesPerAccio As CColForPuntAgrupPerAccio
Private m_ColMomentsPuntualsAgrupatsPerAccio As CColMomPuntAgrupPerAccio
Private m_MostrarEixosLocalsActivat As Boolean
Private m_MostrarConsultaDePefilActivada As Boolean
Private m_Serie As String

Property Get Serie() As String
Serie = m_Serie
End Property
Property Let Serie(ByVal Val As String)
m_Serie = Val
End Property

Property Get MostrarEixosLocalsActivat() As Boolean
MostrarEixosLocalsActivat = m_MostrarEixosLocalsActivat
End Property
Property Let MostrarEixosLocalsActivat(ByVal Val As Boolean)
m_MostrarEixosLocalsActivat = Val
End Property

Property Get MostrarConsultaDePefilActivada() As Boolean
 MostrarConsultaDePefilActivada = m_MostrarConsultaDePefilActivada
End Property
Property Let MostrarConsultaDePefilActivada(ByVal Val As Boolean)
 m_MostrarConsultaDePefilActivada = Val
End Property


Property Get CoefDilatacioTermica() As Double
 CoefDilatacioTermica = m_CoefDilatacioTermica
```

```
End Property
Property Let CoefDilatacioTermica(ByVal Val As Double)
 m_CoefDilatacioTermica = Val
End Property

Property Get Nu() As Double
 Nu = m_Nu
End Property
Property Let Nu(ByVal Val As Double)
 m_Nu = Val
End Property

Property Get Ip() As Double
 Ip = m_Ip
End Property
Property Let Ip(ByVal Val As Double)
 m_Ip = Val
End Property

Property Get G() As Double
 G = m_G
End Property
Property Let G(ByVal Val As Double)
 m_G = Val
End Property

Property Get ColMomentsPuntualsAgrupatsPerAccio() As CColMomPuntAgrupPerAccio
Set ColMomentsPuntualsAgrupatsPerAccio = m_ColMomentsPuntualsAgrupatsPerAccio
End Property
Property Let ColMomentsPuntualsAgrupatsPerAccio(ByVal Val As
CColMomPuntAgrupPerAccio)
Set m_ColMomentsPuntualsAgrupatsPerAccio = Val
End Property

Property Get ColForcesPuntualsAgrupadesPerAccio() As CColForPuntAgrupPerAccio
Set ColForcesPuntualsAgrupadesPerAccio = m_ColForcesPuntualsAgrupadesPerAccio
End Property
Property Let ColForcesPuntualsAgrupadesPerAccio(ByVal Val As
CColForPuntAgrupPerAccio)
Set m_ColForcesPuntualsAgrupadesPerAccio = Val
End Property

Property Get AngleAlfaCREixosLocals() As Double
AngleAlfaCREixosLocals = m_AngleAlfaCREixosLocals
End Property
Property Let AngleAlfaCREixosLocals(ByVal Val As Double)
m_AngleAlfaCREixosLocals = Val
End Property

Property Get ColCarreguesRepartides() As CColCarregaRepartida
Set ColCarreguesRepartides = m_ColCarreguesRepartides
```

```
End Property
Property Let ColCarreguesRepartides(ByVal Val As CColCarregaRepartida)
Set m_ColCarreguesRepartides = Val
End Property

Property Get UnitatsPreferencials() As String
UnitatsPreferencials = m_UnitatsPreferencials
End Property
Property Let UnitatsPreferencials(ByVal Val As String)
m_UnitatsPreferencials = Val
End Property


Property Get RMaterial() As Integer
RMaterial = m_RMaterial
End Property
Property Let RMaterial(ByVal Val As Integer)
m_RMaterial = Val
End Property

Property Get GMaterial() As Integer
GMaterial = m_GMaterial
End Property
Property Let GMaterial(ByVal Val As Integer)
m_GMaterial = Val
End Property

Property Get BMaterial() As Integer
BMaterial = m_BMaterial
End Property
Property Let BMaterial(ByVal Val As Integer)
m_BMaterial = Val
End Property




Property Get OrdreDeDibuixSeleccionat() As Boolean
OrdreDeDibuixSeleccionat = m_OrdreDeDibuixSeleccionat
End Property
Property Let OrdreDeDibuixSeleccionat(ByVal Val As Boolean)
m_OrdreDeDibuixSeleccionat = Val
End Property

Property Get XCentral() As Double
XCentral = (m_X1 + m_X2) / 2
End Property
Property Get YCentral() As Double
YCentral = (m_Y1 + m_Y2) / 2
```

```
End Property
Property Get ZCentral() As Double
ZCentral = (m_Z1 + m_Z2) / 2
End Property

Property Get ProfunditatZ() As Double
ProfunditatZ = m_ProfunditatZ
End Property
Property Let ProfunditatZ(ByVal Val As Double)
m_ProfunditatZ = Val
End Property

Property Get OrdreDeDibuix() As Long
OrdreDeDibuix = m_OrdreDeDibuix
End Property
Property Let OrdreDeDibuix(ByVal Val As Long)
m_OrdreDeDibuix = Val
End Property

Property Get CaresVisibles() As Boolean
CaresVisibles = m_CaresVisibles
End Property
Property Let CaresVisibles(ByVal Val As Boolean)
m_CaresVisibles = Val
End Property

Property Get Material() As String
Material = m_Material
End Property
Property Let Material(ByVal Val As String)
m_Material = Val
End Property

Property Get Densitat() As Double
Densitat = m_Densitat
End Property
Property Let Densitat(ByVal Val As Double)
m_Densitat = Val
End Property

Property Get B() As Double
B = m_B
End Property
Property Let B(ByVal Val As Double)
m_B = Val
End Property

Property Get h() As Double
h = m_H
End Property
Property Let h(ByVal Val As Double)
```

```
m_H = Val
End Property

Property Get R() As Double
R = m_R
End Property
Property Let R(ByVal Val As Double)
m_R = Val
End Property

Property Get DE() As Double
DE = m_DE
End Property
Property Let DE(ByVal Val As Double)
m_DE = Val
End Property

Property Get DE1() As Double
DE1 = m_DE1
End Property
Property Let DE1(ByVal Val As Double)
m_DE1 = Val
End Property

Property Get TipusPerfil() As Integer
TipusPerfil = m_TipusPerfil
End Property
Property Let TipusPerfil(ByVal Val As Integer)
m_TipusPerfil = Val
End Property



Property Get AngleAlfaCREixosGlobals() As Double
AngleAlfaCREixosGlobals = m_AngleAlfaCREixosGlobals
End Property
Property Let AngleAlfaCREixosGlobals(ByVal Val As Double)
m_AngleAlfaCREixosGlobals = Val
End Property

Property Get AngleBetaCREixosGlobals() As Double
AngleBetaCREixosGlobals = m_AngleBetaCREixosGlobals
End Property
Property Let AngleBetaCREixosGlobals(ByVal Val As Double)
m_AngleBetaCREixosGlobals = Val
End Property

Property Get ZCarregaIni() As Double
ZCarregaIni = m_ZCarregaIni
End Property
Property Let ZCarregaIni(ByVal Val As Double)
```

```
m_ZCarregaIni = Val
End Property

Property Get ZCarregaFi() As Double
ZCarregaFi = m_ZCarregaFi
End Property
Property Let ZCarregaFi(ByVal Val As Double)
m_ZCarregaFi = Val
End Property

Property Get CarregaIni() As Double
CarregaIni = m_CarregaIni
End Property
Property Let CarregaIni(ByVal Val As Double)
m_CarregaIni = Val
End Property

Property Get CarregaFi() As Double
CarregaFi = m_CarregaFi
End Property
Property Let CarregaFi(ByVal Val As Double)
m_CarregaFi = Val
End Property


Property Get TipusDeCarregaRepartida() As Integer
TipusDeCarregaRepartida = m_TipusDeCarregaRepartida
End Property
Property Let TipusDeCarregaRepartida(ByVal Val As Integer)
m_TipusDeCarregaRepartida = Val
End Property


Property Get Node1Articulat() As Boolean
Node1Articulat = m_Node1Articulat
End Property
Property Let Node1Articulat(ByVal Val As Boolean)
m_Node1Articulat = Val
End Property

Property Get Node2Articulat() As Boolean
Node2Articulat = m_Node2Articulat
End Property
Property Let Node2Articulat(ByVal Val As Boolean)
m_Node2Articulat = Val
End Property


Property Get AngleOrientacio() As Double
AngleOrientacio = m_AngleOrientacio
End Property
```

```
Property Let AngleOrientacio(ByVal Val As Double)
m_AngleOrientacio = Val
End Property

Property Get Seleccionat() As Boolean
Seleccionat = m_Seleccionat
End Property
Property Let Seleccionat(ByVal Val As Boolean)
m_Seleccionat = Val
End Property

Private Sub CalculaMatriuDeRigidesaLocal()
'Troba totes les matrius de rigidesa i les _
emmagatzema a la col.lecció Matrius

Dim K(1 To 12, 1 To 12) As Double
Dim E As Double
Dim A As Double
Dim L As Double
Dim Ix As Double
Dim Iy As Double
Dim G As Double

   E = m_E
   A = m_A
   L = ((m_X2 - m_X1) ^ 2 + (m_Y2 - m_Y1) ^ 2 + (m_Z2 _
- m_Z1) ^ 2) ^ 0.5
   Ix = m_Ix
   Iy = m_Iy
   G = m_GJ

   'Construïm la matriu de rigidesa 3D per a cada barra
   'Part superior de la matriu com a comentari la posició _
   que ocupen en la pàgina I.26

   K(3, 3) = E * A / L              '1,1
   K(3, 9) = -E * A / L             '1,7

   K(2, 2) = 12 * E * Ix / L ^ 3     '2,2
   K(2, 4) = -6 * E * Ix / L ^ 2     '2,6
   K(2, 8) = -12 * E * Ix / L ^ 3    '2,8
   K(2, 10) = -6 * E * Ix / L ^ 2    '2,12

   K(1, 1) = 12 * E * Iy / L ^ 3     '3,3
   K(1, 5) = 6 * E * Iy / L ^ 2      '3,5
   K(1, 7) = -12 * E * Iy / L ^ 3    '3,9
   K(1, 11) = 6 * E * Iy / L ^ 2     '3,11

   K(6, 6) = G / L                  '4,4
   K(6, 12) = -G / L                '4,10
```

```
    K(5, 5) = 4 * E * Iy / L          '5,5

    K(5, 7) = -6 * E * Iy / L ^ 2     '5,9
    K(5, 11) = 2 * E * Iy / L         '5,11

    K(4, 4) = 4 * E * Ix / L          '6,6
    K(4, 8) = 6 * E * Ix / L ^ 2      '6,8
    K(4, 10) = 2 * E * Ix / L         '6,12

    K(9, 9) = A * E / L               '7,7

    K(8, 8) = 12 * E * Ix / L ^ 3     '8,8
    K(8, 10) = 6 * E * Ix / L ^ 2     '8,12

    K(7, 7) = 12 * E * Iy / L ^ 3     '9,9
    K(7, 11) = -6 * E * Iy / L ^ 2    '9,11

    K(12, 12) = G / L                 '10,10

    K(11, 11) = 4 * E * Iy / L        '11,11

    K(10, 10) = 4 * E * Ix / L        '12,12

    'Part inferior de la matriu

    K(9, 3) = -E * A / L              '1,7

    K(4, 2) = -6 * E * Ix / L ^ 2     '2,6
    K(8, 2) = -12 * E * Ix / L ^ 3    '2,8
    K(10, 2) = -6 * E * Ix / L ^ 2    '2,12

    K(5, 1) = 6 * E * Iy / L ^ 2      '3,5
    K(7, 1) = -12 * E * Iy / L ^ 3    '3,9
    K(11, 1) = 6 * E * Iy / L ^ 2     '3,11

    K(12, 6) = -G / L                 '4,10

    K(7, 5) = -6 * E * Iy / L ^ 2     '5,9
    K(11, 5) = 2 * E * Iy / L         '5,11

    K(8, 4) = 6 * E * Ix / L ^ 2      '6,8
    K(10, 4) = 2 * E * Ix / L         '6,12

    K(10, 8) = 6 * E * Ix / L ^ 2     '8,12

    K(11, 7) = -6 * E * Iy / L ^ 2    '9,11

    m_MatriuDeRigidesaLocal = K

End Sub
```

```
Property Get ColorTextBarra() As Double
ColorTextBarra = m_ColorTextBarra
End Property
Property Get MatriudeRigidesaGlobalElemental() As Variant

    CalculaMatriuDeRigidesaGlobalElemental
    MatriudeRigidesaGlobalElemental = m_MatriuDeRigidesaGlobalElemental

End Property
Property Get VectorCorriments() As Variant
VectorCorriments = m_VectorCorriments
End Property
Property Let VectorCorriments(ByVal Val As Variant)
m_VectorCorriments = Val
End Property
Property Get MatriuDeRigidesaGlobalCalculat() As Boolean
MatriuDeRigidesaGlobalCalculat = m_MatriuDeRigidesaGlobalCalculat
End Property

Property Get Numero() As Long
Numero = m_Numero
End Property
Property Get NumNode1() As Long
NumNode1 = m_NumNode1
End Property
Property Get NumNode2() As Long
NumNode2 = m_NumNode2
End Property
Property Get X1() As Double
X1 = m_X1
End Property
Property Get X2() As Double
X2 = m_X2
End Property
Property Get Y1() As Double
Y1 = m_Y1
End Property
Property Get Y2() As Double
Y2 = m_Y2
End Property
Property Get Z1() As Double
Z1 = m_Z1
End Property
Property Get Z2() As Double
Z2 = m_Z2
End Property


Property Get E() As Double
E = m_E
End Property
```

```
Property Get Ix() As Double
Ix = m_Ix
End Property
Property Get Iy() As Double
Iy = m_Iy
End Property

Property Get L() As Double
L = ((m_X2 - m_X1) ^ 2 + (m_Y2 - m_Y1) ^ 2 + (m_Z2 _
- m_Z1) ^ 2) ^ 0.5
End Property

Property Get GJ() As Double
GJ = m_GJ
End Property
Property Get A() As Double
A = m_A
End Property
Property Get Nom() As String
Nom = m_Nom
End Property
Property Get Denominacio() As String
Denominacio = m_Denominacio
End Property

Property Get VxOrto() As Double

Dim VD(1 To 3) As Double
Dim VO(1 To 3) As Double
Dim VN(1 To 3) As Double
Dim PuntPrima(1 To 3, 1 To 1) As Double
Dim M(1 To 3, 1 To 3) As Double
Dim Modul As Double


   VD(1) = m_X2 - m_X1
   VD(2) = m_Y2 - m_Y1
   VD(3) = m_Z2 - m_Z1
   Modul = (VD(1) ^ 2 + VD(2) ^ 2 + VD(3) ^ 2) ^ 0.5
   VD(1) = VD(1) / Modul: VD(2) = VD(2) / Modul: VD(3) = VD(3) / Modul
   Select Case EsUnPilar
   Case False
   VO(1) = VD(3)
   VO(2) = 0
   VO(3) = -VD(1)
   Case True
   VO(1) = 0
   VO(2) = VD(3)
   VO(3) = -VD(2)
   End Select
```

```
    VN(1) = VD(2) * VO(3) - VD(3) * VO(2)
    VN(2) = VD(3) * VO(1) - VD(1) * VO(3)
    VN(3) = VD(1) * VO(2) - VD(2) * VO(1)

    PuntPrima(1, 1) = Cos(m_AngleOrientacio)
    PuntPrima(2, 1) = -Sin(m_AngleOrientacio)
    PuntPrima(3, 1) = 0

    M(1, 1) = VO(1): M(2, 1) = VO(2): M(3, 1) = VO(3)
    M(1, 2) = VN(1): M(2, 2) = VN(2): M(3, 2) = VN(3)
    M(1, 3) = VD(1): M(2, 3) = VD(2): M(3, 3) = VD(3)

    'Multiplico les matrius M i Puntprima a pic i pala
    VxOrto = M(1, 1) * PuntPrima(1, 1) + _
    M(1, 2) * PuntPrima(2, 1) + M(1, 3) * PuntPrima(3, 1)

End Property
Property Get VyOrto() As Double

Dim VD(1 To 3) As Double
Dim VO(1 To 3) As Double
Dim VN(1 To 3) As Double
Dim PuntPrima(1 To 3, 1 To 1) As Double
Dim Res() As Double
Dim M(1 To 3, 1 To 3) As Double
Dim Modul As Double


    VD(1) = m_X2 - m_X1
    VD(2) = m_Y2 - m_Y1
    VD(3) = m_Z2 - m_Z1
    Modul = (VD(1) ^ 2 + VD(2) ^ 2 + VD(3) ^ 2) ^ 0.5
    VD(1) = VD(1) / Modul: VD(2) = VD(2) / Modul: VD(3) = VD(3) / Modul
    Select Case EsUnPilar
    Case False
    VO(1) = VD(3)
    VO(2) = 0
    VO(3) = -VD(1)
    Case True
    VO(1) = 0
    VO(2) = VD(3)
    VO(3) = -VD(2)
    End Select

    VN(1) = VD(2) * VO(3) - VD(3) * VO(2)
    VN(2) = VD(3) * VO(1) - VD(1) * VO(3)
    VN(3) = VD(1) * VO(2) - VD(2) * VO(1)

    PuntPrima(1, 1) = Cos(m_AngleOrientacio)
    PuntPrima(2, 1) = -Sin(m_AngleOrientacio)
    PuntPrima(3, 1) = 0
```

```
    M(1, 1) = VO(1): M(2, 1) = VO(2): M(3, 1) = VO(3)
    M(1, 2) = VN(1): M(2, 2) = VN(2): M(3, 2) = VN(3)
    M(1, 3) = VD(1): M(2, 3) = VD(2): M(3, 3) = VD(3)

    VyOrto = M(2, 1) * PuntPrima(1, 1) + _
    M(2, 2) * PuntPrima(2, 1) + M(2, 3) * PuntPrima(3, 1)



End Property

Property Get VzOrto() As Double

Dim VD(1 To 3) As Double
Dim VO(1 To 3) As Double
Dim VN(1 To 3) As Double
Dim PuntPrima(1 To 3, 1 To 1) As Double
Dim Res() As Double
Dim M(1 To 3, 1 To 3) As Double
Dim Modul As Double

      VD(1) = m_X2 - m_X1
    VD(2) = m_Y2 - m_Y1
    VD(3) = m_Z2 - m_Z1
    Modul = (VD(1) ^ 2 + VD(2) ^ 2 + VD(3) ^ 2) ^ 0.5
    VD(1) = VD(1) / Modul: VD(2) = VD(2) / Modul: VD(3) = VD(3) / Modul
    Select Case EsUnPilar
    Case False
    VO(1) = VD(3)
    VO(2) = 0
    VO(3) = -VD(1)
    Case True
    VO(1) = 0
    VO(2) = VD(3)
    VO(3) = -VD(2)
    End Select

    VN(1) = VD(2) * VO(3) - VD(3) * VO(2)
    VN(2) = VD(3) * VO(1) - VD(1) * VO(3)
    VN(3) = VD(1) * VO(2) - VD(2) * VO(1)

    PuntPrima(1, 1) = Cos(m_AngleOrientacio)
    PuntPrima(2, 1) = -Sin(m_AngleOrientacio)
    PuntPrima(3, 1) = 0

    M(1, 1) = VO(1): M(2, 1) = VO(2): M(3, 1) = VO(3)
    M(1, 2) = VN(1): M(2, 2) = VN(2): M(3, 2) = VN(3)
    M(1, 3) = VD(1): M(2, 3) = VD(2): M(3, 3) = VD(3)

    VzOrto = M(3, 1) * PuntPrima(1, 1) + _
```

```
    M(3, 2) * PuntPrima(2, 1) + M(3, 3) * PuntPrima(3, 1)


End Property

Property Get FxNode1() As Double
FxNode1 = m_FxNode1
End Property
Property Get FyNode1() As Double
FyNode1 = m_FyNode1
End Property
Property Get FzNode1() As Double
FzNode1 = m_FzNode1
End Property
Property Get MxNode1() As Double
MxNode1 = m_MxNode1
End Property
Property Get MyNode1() As Double
MyNode1 = m_MyNode1
End Property
Property Get MzNode1() As Double
MzNode1 = m_MzNode1
End Property
Property Get FxNode2() As Double
FxNode2 = m_FxNode2
End Property
Property Get FyNode2() As Double
FyNode2 = m_FyNode2
End Property
Property Get FzNode2() As Double
FzNode2 = m_FzNode2
End Property
Property Get MxNode2() As Double
MxNode2 = m_MxNode2
End Property
Property Get MyNode2() As Double
MyNode2 = m_MyNode2
End Property
Property Get MzNode2() As Double
MzNode2 = m_MzNode2
End Property
'Propietats Let de CBarra
Property Let X1(ByVal Val As Double)
m_X1 = Val
End Property
Property Let X2(ByVal Val As Double)
m_X2 = Val
End Property
Property Let Y1(ByVal Val As Double)
m_Y1 = Val
End Property
```

```
Property Let Y2(ByVal Val As Double)
m_Y2 = Val
End Property
Property Let Z1(ByVal Val As Double)
m_Z1 = Val
End Property
Property Let Z2(ByVal Val As Double)
m_Z2 = Val
End Property


Property Let Numero(Val As Long)
m_Numero = Val
End Property
Property Let NumNode1(ByVal Val As Long)
m_NumNode1 = Val
End Property
Property Let NumNode2(ByVal Val As Long)
m_NumNode2 = Val
End Property
Property Let E(ByVal Val As Double)
m_E = Val
End Property
Property Let Ix(ByVal Val As Double)
m_Ix = Val
End Property
Property Let Iy(ByVal Val As Double)
m_Iy = Val
End Property
Property Let GJ(ByVal Val As Double)
m_GJ = Val
End Property
Property Let A(ByVal Val As Double)
m_A = Val
End Property
Property Let Nom(ByVal Val As String)
m_Nom = Val
End Property
Property Let Denominacio(ByVal Val As String)
m_Denominacio = Val
End Property
Property Let FxNode1(ByVal Val As Double)
m_FxNode1 = Val
End Property
Property Let FyNode1(ByVal Val As Double)
m_FyNode1 = Val
End Property
Property Let FzNode1(ByVal Val As Double)
m_FzNode1 = Val
End Property
Property Let MxNode1(ByVal Val As Double)
```

```
m_MxNode1 = Val
End Property
Property Let MyNode1(ByVal Val As Double)
m_MyNode1 = Val
End Property
Property Let MzNode1(ByVal Val As Double)
m_MzNode1 = Val
End Property
Property Let FxNode2(ByVal Val As Double)
m_FxNode2 = Val
End Property
Property Let FyNode2(ByVal Val As Double)
m_FyNode2 = Val
End Property
Property Let FzNode2(ByVal Val As Double)
m_FzNode2 = Val
End Property
Property Let MxNode2(ByVal Val As Double)
m_MxNode2 = Val
End Property
Property Let MyNode2(ByVal Val As Double)
m_MyNode2 = Val
End Property
Property Let MzNode2(ByVal Val As Double)
m_MzNode2 = Val
End Property
Property Let ColorTextBarra(ByVal Val As Double)
m_ColorTextBarra = Val
End Property

Private Sub CalculaMatriuRotacional()
'Primer trobem la matriu rotacional

'Segons la figura I.2.8 del Tomo I de Teoria _
D'Estructures
Dim lx As Double
Dim Mx As Double
Dim nx As Double

Dim ly As Double
Dim My As Double
Dim ny As Double

Dim lz As Double
Dim Mz As Double
Dim nz As Double

Dim vx1 As Double
Dim vx2 As Double
Dim vx3 As Double
```

```
Dim vy1 As Double
Dim vy2 As Double
Dim vy3 As Double

Dim vz1 As Double
Dim vz2 As Double
Dim vz3 As Double

Dim Modul As Double
Dim NumNode1 As Long
Dim NumNode2 As Long
Dim Alfa As Double
Dim Lambda(1 To 12, 1 To 12)


    NumNode1 = m_NumNode1
    NumNode2 = m_NumNode2
    vz1 = m_X2 - m_X1
    vz2 = m_Y2 - m_Y1
    vz3 = m_Z2 - m_Z1

    'Normalitzo
    Modul = (vz1 ^ 2 + vz2 ^ 2 + vz3 ^ 2) ^ 0.5
    vz1 = vz1 / Modul: vz2 = vz2 / Modul: vz3 = vz3 / Modul
    vx1 = m_VxOrto
    vx2 = m_VyOrto
    vx3 = m_VzOrto

    'Normalitzo
    Modul = (vx1 ^ 2 + vx2 ^ 2 + vx3 ^ 2) ^ 0.5
    vx1 = vx1 / Modul: vx2 = vx2 / Modul: vx3 = vx3 / Modul

    'vy l'obtinc mitjançant el producte vectorial dels _
    vectors vz i vx (en aquest mateix ordre)
    vy1 = vz2 * vx3 - vz3 * vx2
    vy2 = vz3 * vx1 - vz1 * vx3
    vy3 = vz1 * vx2 - vz2 * vx1

    lx = vx1
    ly = vx2
    lz = vx3

    Mx = vy1
    My = vy2
    Mz = vy3

    nx = vz1
    ny = vz2
    nz = vz3

    m_MatriuRotacional = Lambda
```

```
    m_MatriuRotacional(1, 1) = lx: m_MatriuRotacional(1, 2) = ly: m_MatriuRotacional(1, 3) =
lz
    m_MatriuRotacional(2, 1) = Mx: m_MatriuRotacional(2, 2) = My: m_MatriuRotacional(2,
3) = Mz
    m_MatriuRotacional(3, 1) = nx: m_MatriuRotacional(3, 2) = ny: m_MatriuRotacional(3, 3)
= nz

    m_MatriuRotacional(4, 4) = lx: m_MatriuRotacional(4, 5) = ly: m_MatriuRotacional(4, 6) =
lz
    m_MatriuRotacional(5, 4) = Mx: m_MatriuRotacional(5, 5) = My: m_MatriuRotacional(5,
6) = Mz
    m_MatriuRotacional(6, 4) = nx: m_MatriuRotacional(6, 5) = ny: m_MatriuRotacional(6, 6)
= nz

    m_MatriuRotacional(7, 7) = lx: m_MatriuRotacional(7, 8) = ly: m_MatriuRotacional(7, 9) =
lz
    m_MatriuRotacional(8, 7) = Mx: m_MatriuRotacional(8, 8) = My: m_MatriuRotacional(8,
9) = Mz
    m_MatriuRotacional(9, 7) = nx: m_MatriuRotacional(9, 8) = ny: m_MatriuRotacional(9, 9)
= nz

    m_MatriuRotacional(10, 10) = lx: m_MatriuRotacional(10, 11) = ly:
m_MatriuRotacional(10, 12) = lz
    m_MatriuRotacional(11, 10) = Mx: m_MatriuRotacional(11, 11) = My:
m_MatriuRotacional(11, 12) = Mz
    m_MatriuRotacional(12, 10) = nx: m_MatriuRotacional(12, 11) = ny:
m_MatriuRotacional(12, 12) = nz

End Sub

Private Sub CalculaMatriuDeRigidesaGlobalElemental()

Dim Kaux1() As Double
Dim Kaux2() As Double
Dim Kaux3() As Double
Dim Kaux4() As Double
Dim K() As Double

CalculaMatriuDeRigidesaLocal
CalculaMatriuRotacional

    AssignaVariantAMatriu m_MatriuDeRigidesaLocal, Kaux1
    AssignaVariantAMatriu m_MatriuRotacional, Kaux2
    MultiplicaMatrius Kaux1, Kaux2, Kaux3
    TrasposaMatriu Kaux2, Kaux4
    MultiplicaMatrius Kaux4, Kaux3, K
    m_MatriuDeRigidesaGlobalElemental = K
    m_MatriuDeRigidesaGlobalCalculat = True

End Sub
```

```
Private Sub InversaMatriu(MatriuOrigen() As Double, MatriuDesti() As Double)

Dim Ix As Long
Dim jX As Long
Dim iX2 As Long
Dim jX2 As Long
Dim A As Long
Dim Apuja As Long
Dim Trobat As Boolean
Dim Matraux() As Double
Dim Ks As Double
Dim Ki As Double
Dim aux1 As Double
Dim Aux2 As Double
Dim P As Long
Dim Rang As Long
Dim i As Long
Dim j As Long
Dim Origen() As Double

'On Error GoTo Error

Rang = UBound(MatriuOrigen, 1)

ReDim Matraux(1 To Rang, 1 To Rang)
ReDim Origen(1 To Rang, 1 To Rang)
ReDim MatriuDesti(1 To Rang, 1 To Rang)
'Assignem a la matriu Origen la matriu MatriuOrigen per _
evitar modificar la matriu d'entrada

For i = 1 To Rang

   For j = 1 To Rang

   Origen(i, j) = MatriuOrigen(i, j)

   Next j

Next i

'crea la matriu identitat nxn

For Ix = 1 To Rang

   For jX = 1 To Rang

   If Ix = jX Then
   Matraux(Ix, jX) = 1
   End If

   Next jX
```

```
Next Ix

'Triangulació inferior

For jX = 1 To Rang

        'per evitar dividir per zero
        Apuja = jX
        A = jX

           'While ((Origen(jX, jX) = 0) And (Apuja + jX <= Rang) And (Trobat = False))
           While ((Origen(jX, jX) = 0) And (Apuja <= Rang) And (Trobat = False))

              If Origen(A, jX) <> 0 Then
              Apuja = A
              Trobat = True
              Else: A = A + 1

              End If

           Wend

              For P = 1 To Rang

              aux1 = Origen(Apuja, P)
              Aux2 = Matraux(Apuja, P)
              Origen(Apuja, P) = Origen(jX, P)
              Matraux(Apuja, P) = Matraux(jX, P)
              Origen(jX, P) = aux1
              Matraux(jX, P) = Aux2

              Next P

        'fi per evitar dividir per zero
        'això és nou
        Trobat = False
    For iX2 = jX + 1 To Rang

        Ks = Origen(jX, jX)
        Ki = Origen(iX2, jX)

        For jX2 = 1 To Rang

           Origen(iX2, jX2) = Origen(iX2, jX2) - Origen(jX, jX2) * Ki / Ks
           Matraux(iX2, jX2) = Matraux(iX2, jX2) - Matraux(jX, jX2) * Ki / Ks

        Next jX2

    Next iX2
```

```
Next jX

'Triangulació superior

For jX = Rang To 2 Step -1

        'per evitar dividir per zero

        Apuja = jX
        A = jX

          While (Origen(jX, jX) = 0) And (Apuja + jX > 0) And (Trobat = False)

            If Origen(A, jX) <> 0 Then

            Apuja = A
            Trobat = True
            Else: A = A - 1

            End If

          Wend

            For P = 1 To Rang

            aux1 = Origen(Apuja, P)
            Aux2 = Matraux(Apuja, P)
            Origen(Apuja, P) = Origen(jX, P)
            Matraux(Apuja, P) = Matraux(jX, P)
            Origen(jX, P) = aux1
            Matraux(jX, P) = Aux2

            Next P

        'fi per evitar dividir per zero

        For iX2 = jX - 1 To 1 Step -1

            Ki = Origen(jX, jX)
            Ks = Origen(iX2, jX)

            For jX2 = Rang To 1 Step -1

              Origen(iX2, jX2) = Origen(iX2, jX2) - Origen(jX, jX2) * Ks / Ki
              Matraux(iX2, jX2) = Matraux(iX2, jX2) - Matraux(jX, jX2) * Ks / Ki

            Next jX2

        Next iX2

  Next jX
```

```
'Correcció de la Diagonal

For Ix = 1 To Rang

      For jX = 1 To Rang

      Matraux(Ix, jX) = Matraux(Ix, jX) / Origen(Ix, Ix)

      Next jX

Next Ix

'Assignem matraux a K

For i = 1 To Rang

   For j = 1 To Rang

   MatriuDesti(i, j) = Matraux(i, j)

   Next j

Next i

Exit Sub

'Error:
'Call MsgBox(Err.Description, , "Matriu no invertible")

End Sub



 Private Sub TrasposaMatriu(MatriuOrigen() As Double, MatriuDesti() As Double)

   Dim i As Integer
   Dim j As Integer
   Dim Origen() As Double

   ReDim Origen(1 To UBound(MatriuOrigen, 1), 1 To UBound(MatriuOrigen, 2))
   ReDim MatriuDesti(1 To UBound(Origen, 2), 1 To UBound(Origen, 1))

   For i = 1 To UBound(Origen, 1)

      For j = 1 To UBound(Origen, 2)
      Origen(i, j) = MatriuOrigen(i, j)
      Next j

   Next i
```

```
  For i = LBound(Origen, 1) To UBound(Origen, 1)

      For j = LBound(Origen, 2) To UBound(Origen, 2)
          MatriuDesti(j, i) = Origen(i, j)
      Next j

  Next i

End Sub

Private Sub AssignaVariantAMatriu(Origen As Variant, Desti() As Double)

Dim imin As Long
Dim imax As Long
Dim i As Long
Dim j As Long

imax = UBound(Origen, 1)
imin = LBound(Origen, 1)

ReDim Desti(imin To imax, imin To imax)

   For i = imin To imax

      For j = imin To imax

      Desti(i, j) = Origen(i, j)

      Next j

   Next i

End Sub

Private Sub MultiplicaMatrius(A() As Double, B() As Double, _
 C() As Double)

   Dim AiMin As Long, AiMax As Long
   Dim AjMin As Long, AjMax As Long
   Dim BiMin As Long, BiMax As Long
   Dim BjMin As Long, BjMax As Long
   Dim CiMin As Long, CiMax As Long
   Dim CjMin As Long, CjMax As Long
   Dim Ai As Long, Bi As Long, Ci As Long
   Dim Aj As Long, Bj As Long, Cj As Long

   AiMin = LBound(A, 1)
   AjMin = LBound(A, 2)
   BiMin = LBound(B, 1)
   BjMin = LBound(B, 2)
```

```
    AiMax = UBound(A, 1)
    AjMax = UBound(A, 2)
    BiMax = UBound(B, 1)
    BjMax = UBound(B, 2)

    ReDim C(AiMin To AiMax, BjMin To BjMax)

    CiMin = LBound(C, 1)
    CjMin = LBound(C, 2)
    CiMax = UBound(C, 1)
    CjMax = UBound(C, 2)

    For Ci = CiMin To CiMax

        For Cj = CjMin To CjMax

            For Aj = AjMin To AjMax

            C(Ci, Cj) = C(Ci, Cj) + _
            A(Ci, Aj) * B(Aj, Cj)

            Next Aj

        Next Cj

    Next Ci

End Sub

Property Get EsUnPilar() As Boolean

If (m_X1 = m_X2) And (m_Z1 = m_Z2) Then
EsUnPilar = True
Else:
EsUnPilar = False
End If

End Property



Function ExisteixLaAccioRepartida(Accio As Long) As Boolean

Dim AccioEnCurs As Long, MaxAccio As Long

    If m_ColCarreguesRepartides.Count = 0 Then
    ExisteixLaAccioRepartida = False
    Exit Function
    End If

    MaxAccio = m_ColCarreguesRepartides.Count
```

```
    For AccioEnCurs = 1 To MaxAccio
      If Accio = m_ColCarreguesRepartides.Item(AccioEnCurs).NumeroDeCarrega Then
      ExisteixLaAccioRepartida = True
      Exit Function
      End If
    Next AccioEnCurs

End Function


Function ExisteixLaColeccioDeForcesNodalsAgrupadesPerAccio(Accio As Long) As
Boolean

Dim AccioEnCurs As Long, MaxAccio As Long

   If m_ColForcesPuntualsAgrupadesPerAccio.Count = 0 Then
   ExisteixLaColeccioDeForcesNodalsAgrupadesPerAccio = False
   Exit Function
   End If

   MaxAccio = m_ColForcesPuntualsAgrupadesPerAccio.Count

    For AccioEnCurs = 1 To MaxAccio
      If Accio =
m_ColForcesPuntualsAgrupadesPerAccio.Item(AccioEnCurs).NumeroDeAccio Then
      ExisteixLaColeccioDeForcesNodalsAgrupadesPerAccio = True
      Exit Function
      End If
    Next AccioEnCurs

End Function


Private Sub Class_Initialize()

Set m_ColForcesPuntualsAgrupadesPerAccio = New CColForPuntAgrupPerAccio
Set m_ColCarreguesRepartides = New CColCarregaRepartida
End Sub

Private Sub Class_Terminate()
Set m_ColForcesPuntualsAgrupadesPerAccio = Nothing
Set m_ColCarreguesRepartides = Nothing
End Sub
```

C. Classe Cnode

```
Option Explicit

Private m_X As Double
Private m_Y As Double
```

```
Private m_Z As Double
Private m_MomentX As Double
Private m_MomentY As Double
Private m_MomentZ As Double
Private m_Fx As Double
Private m_Fy As Double
Private m_Fz As Double
Private m_XBlock As Boolean
Private m_YBlock As Boolean
Private m_ZBlock As Boolean
Private m_MXBlock As Boolean
Private m_MYBlock As Boolean
Private m_MZBlock As Boolean
Private m_Numero As Long
Private m_ColorTextNode As Double
Private m_ColorTextNodeFormularis As Long
Private m_Seleccionat As Boolean
Private m_ColSolicitacioNodal As CColSolicitacioNodal


Property Get ColSolicitacioNodal() As CColSolicitacioNodal
Set ColSolicitacioNodal = m_ColSolicitacioNodal
End Property

Property Let ColSolicitacioNodal(ByVal Val As CColSolicitacioNodal)
Set m_ColSolicitacioNodal = Val
End Property

Property Get ColorTextNodeFormularis() As Long
ColorTextNodeFormularis = m_ColorTextNodeFormularis
End Property

Property Let ColorTextNodeFormularis(ByVal Val As Long)
m_ColorTextNodeFormularis = Val
End Property


Property Get Seleccionat() As Boolean
Seleccionat = m_Seleccionat
End Property

Property Get ColorTextNode() As Double
ColorTextNode = m_ColorTextNode
End Property

Property Let ColorTextNode(ByVal Val As Double)
m_ColorTextNode = Val
End Property

Property Get X() As Double
X = m_X
```

```
End Property

Property Get Y() As Double
Y = m_Y
End Property

Property Get Z() As Double
Z = m_Z
End Property

Property Get MomentX() As Double
MomentX = m_MomentX
End Property

Property Get MomentY() As Double
MomentY = m_MomentY
End Property

Property Get MomentZ() As Double
MomentZ = m_MomentZ
End Property

Property Get Fx() As Double
Fx = m_Fx
End Property

Property Get Fy() As Double
Fy = m_Fy
End Property

Property Get Fz() As Double
Fz = m_Fz
End Property

Property Get Numero() As Double
Numero = m_Numero
End Property

Property Get XBlock() As Boolean
XBlock = m_XBlock
End Property

Property Get YBlock() As Boolean
YBlock = m_YBlock
End Property

Property Get ZBlock() As Boolean
ZBlock = m_ZBlock
End Property

Property Get MXBlock() As Boolean
```

```
MXBlock = m_MXBlock
End Property

Property Get MYBlock() As Boolean
MYBlock = m_MYBlock
End Property

Property Get MZBlock() As Boolean
MZBlock = m_MZBlock
End Property

'Propietats Let de Node
Property Let X(ByVal Val As Double)
m_X = Val
End Property

Property Let Y(ByVal Val As Double)
m_Y = Val
End Property

Property Let Z(ByVal Val As Double)
m_Z = Val
End Property

Property Let MomentX(ByVal Val As Double)
m_MomentX = Val
End Property

Property Let MomentY(ByVal Val As Double)
m_MomentY = Val
End Property

Property Let MomentZ(ByVal Val As Double)
m_MomentZ = Val
End Property

Property Let Fx(ByVal Val As Double)
m_Fx = Val
End Property

Property Let Fy(ByVal Val As Double)
m_Fy = Val
End Property

Property Let Fz(ByVal Val As Double)
m_Fz = Val
End Property

Property Let Numero(ByVal Val As Double)
m_Numero = Val
End Property
```

```
Property Let XBlock(ByVal Val As Boolean)
m_XBlock = Val
End Property

Property Let YBlock(ByVal Val As Boolean)
m_YBlock = Val
End Property

Property Let ZBlock(ByVal Val As Boolean)
m_ZBlock = Val
End Property

Property Let MXBlock(ByVal Val As Boolean)
m_MXBlock = Val
End Property

Property Let MYBlock(ByVal Val As Boolean)
m_MYBlock = Val
End Property

Property Let MZBlock(ByVal Val As Boolean)
m_MZBlock = Val
End Property

Property Let Seleccionat(ByVal Val As Boolean)
m_Seleccionat = Val
End Property

Private Sub Class_Initialize()
Set m_ColSolicitacioNodal = New CColSolicitacioNodal
End Sub

Private Sub Class_Terminate()
Set m_ColSolicitacioNodal = Nothing
End Sub

Function ExisteixLaAccioNodal(Accio As Long) As Boolean

Dim AccioEnCurs As Long, MaxAccio As Long

   If m_ColSolicitacioNodal.Count = 0 Then
   ExisteixLaAccioNodal = False
   Exit Function
   End If

   MaxAccio = m_ColSolicitacioNodal.Count

      For AccioEnCurs = 1 To MaxAccio
         If Accio = m_ColSolicitacioNodal.Item(AccioEnCurs).NumeroDeCarrega Then
         ExisteixLaAccioNodal = True
```

```
            Exit Function
            End If
        Next AccioEnCurs

End Function
```

II. Llistat de mòduls

A. Accés a la base de dades

```
Option Explicit


Sub DesconnectarBaseDeDades()
Principal.EsborrarLEstructura
'CREO UN ACCES A LA BASE DE DADES
'CREO DOS NODES
EstructuraVis.ColNode.Add 1, 0, 0, 0, False, False, False, False, False, False, 0, 0, 0, 0, 0,
0, 0, False
EstructuraVis.ColNode.Add 2, 1, 0, 0, False, False, False, False, False, False, 0, 0, 0, 0, 0,
0, 0, False
'CREO UNA BARRA
IntroBarres.DeterminaBarraEnCurs
EstructuraVis.ColBarra.Add 1, 1, 2, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1
Load frmTipusBarra
'EM DESLLIGO DE LA BASE DE DADES
frmTipusBarra.Adodc1.ConnectionString = ""
frmTipusBarra.Adodc2.ConnectionString = ""
Unload frmTipusBarra
'ESBORRO LA BARRA
EstructuraVis.ColBarra.Remove (1)
'ESBORRO ELS DOS NODES
EstructuraVis.ColNode.Remove (1)
EstructuraVis.ColNode.Remove (1)
IntroBarres.DeterminaBarraEnCurs
IntrodNodes.DeterminaNodeEnCurs
End Sub

Sub DeterminaTipusPerfil()
Dim BarraEnCurs As Long
BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs



With EstructuraVis.ColBarra.Item(Str(BarraEnCurs))
Select Case frmTipusBarra.Combo1.Text

Case "Sèrie IPN"
.TipusPerfil = 3
Case "Sèrie IPE"
.TipusPerfil = 3
Case "Sèrie HEB"
.TipusPerfil = 3
```

```
Case "Sèrie HEA"
.TipusPerfil = 3
Case "Sèrie HEM"
.TipusPerfil = 3
Case "Sèrie UPN"
.TipusPerfil = 4
Case "Sèrie L"
.TipusPerfil = 5
Case "Buits Rodons"
.TipusPerfil = 6
Case "Buits Quadrats"
.TipusPerfil = 7
Case "Buits Rectangulars"
.TipusPerfil = 8
Case "Usuari"
.TipusPerfil = 9
End Select

.Material = EstructuraVis.Preferencies.MaterialPerDefecteSeccionsMetaliques
End With
End Sub

Function ElNomDeMaterialJaExisteixONoEsValid(NomMaterial As String) As Boolean
Dim i As Long, imax As Long



With frmTipusBarra.Adodc2
imax = .Recordset.RecordCount

   For i = 1 To imax
      If .Recordset.EOF = True Then .Recordset.MoveFirst
      If .Recordset(1) = NomMaterial Then
      ElNomDeMaterialJaExisteixONoEsValid = True
      MsgBox "Ja existeix un material referenciat amb aquest nom", vbInformation
      Exit Function
      End If
      .Recordset.MoveNext
   Next i


End With
      If NomMaterial = "" Then
      ElNomDeMaterialJaExisteixONoEsValid = True
      MsgBox "Nom de material no vàlid", vbInformation
      Exit Function
      End If


ElNomDeMaterialJaExisteixONoEsValid = False
```

```
End Function


Function ElNomDePerfilJaExisteixONoEsValid(NomPerfil As String) As Boolean
Dim i As Long, imax As Long
Dim Auxiliar As String


With frmTipusBarra.Adodc1
Auxiliar = .RecordSource
.RecordSource = "select * from perfilsdus"
.Refresh
imax = .Recordset.RecordCount

   For i = 1 To imax
      If .Recordset.EOF = True Then .Recordset.MoveFirst
      If .Recordset(1) = NomPerfil Then

      ElNomDePerfilJaExisteixONoEsValid = True
      MsgBox "Ja existeix una secció referenciada amb aquest nom", vbInformation
      Exit Function
      End If
      .Recordset.MoveNext
   Next i


.RecordSource = Auxiliar
End With
      If NomPerfil = "" Then
      ElNomDePerfilJaExisteixONoEsValid = True
      MsgBox "Nom de la secció no és vàlid", vbInformation
      Exit Function
      End If


ElNomDePerfilJaExisteixONoEsValid = False

End Function


Sub OmpleBarraAuxiliar()
Dim BarraEnCurs As Long

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs

With EstructuraVis.AuxFormularis
.BarraAuxiliar = EstructuraVis.ColBarra.Item(Str(BarraEnCurs))
End With
```

```
End Sub


Sub OmpleBarraAEditar()
Dim BarraAEditar As Long
Dim BarraEnCurs As Long

BarraEnCurs = EstructuraVis.AuxFormularis.BarraAEditar

With EstructuraVis.AuxFormularis
.BarraAuxiliar = EstructuraVis.ColBarra.Item(Str(BarraEnCurs))
End With
End Sub



Sub CarregaBarraAuxiliar()
Dim BarraEnCurs As Long

If EstructuraVis.ColBarra.Count < 1 Then Exit Sub

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs


With EstructuraVis.AuxFormularis
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).A = .BarraAuxiliar.A
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).AngleOrientacio =
.BarraAuxiliar.AngleOrientacio
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).B = .BarraAuxiliar.B
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).h = .BarraAuxiliar.h
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).R = .BarraAuxiliar.R
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).CoefDilatacioTermica =
.BarraAuxiliar.CoefDilatacioTermica
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).DE = .BarraAuxiliar.DE
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).DE1 = .BarraAuxiliar.DE1
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).G = .BarraAuxiliar.G
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).GJ = .BarraAuxiliar.GJ
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil = .BarraAuxiliar.TipusPerfil
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Nom = .BarraAuxiliar.Nom
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Denominacio = .BarraAuxiliar.Denominacio
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Ip = .BarraAuxiliar.Ip
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Ix = .BarraAuxiliar.Ix
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Iy = .BarraAuxiliar.Iy
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Nu = .BarraAuxiliar.Nu
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Serie = .BarraAuxiliar.Serie
If .BarraAuxiliar.Material = "" Then
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Material = "Formigó"
Else
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Material = .BarraAuxiliar.Material
```

```
End If
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Node1Articulat =
.BarraAuxiliar.Node1Articulat
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Node2Articulat =
.BarraAuxiliar.Node2Articulat


End With

End Sub
Sub TFPGotFocus()
Screen.MousePointer = vbDefault
AccesADades.CarregaBarraAuxiliar
AccesADades.RedibuixaFormulariTipusBarra
'DibuixaFinestreta3FTP
End Sub


Sub EditarBaseDeDades()
Dim hwndact As Long, vr As Long
Dim aplic As String, X As String
On Error GoTo Error

aplic = "c:\EstruWin3d\perfils2.mdb"
'aplic = App.Path & "\Perfils2.mdb"
Unload frmTipusBarra
If ShellExecute(hwndact, vbNullString, aplic, vbNullString, _
vbNullString, 1) <= 32 Then
End If
Exit Sub

Error:
MsgBox Err.Description, vbCritical, "No es pot carregar la base de dades"

End Sub

Sub TornaInvisibleFTP()
'frmTipusBarra.Visible = False
Unload frmTipusBarra
Principal.AssignaPunterDelMouseEscaient
End Sub

Sub ValidaCheckArticulacio(Node As Integer)
Dim BarraEnCurs As Long

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs

With EstructuraVis.ColBarra.Item(Str(BarraEnCurs))

Select Case Node
```

```
Case 1
   Select Case frmTipusBarra.Check1.Value
   Case 0
   .Node1Articulat = False
   Case 1
   .Node1Articulat = True
   End Select
Case 2
   Select Case frmTipusBarra.Check2.Value
   Case 0
   .Node2Articulat = False
   Case 1
   .Node2Articulat = True
   End Select
End Select
End With

AccesADades.RedibuixaFormulariTipusBarra

End Sub

Sub ValidaCombo2()
Dim Criteri As String
Dim BarraEnCurs As Long
On Error Resume Next

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs


With frmTipusBarra
   .DataCombo1.DataField = ""
   .Adodc1.Recordset.MoveNext
   Criteri = "Nom LIKE " & "'" & .DataCombo1.Text & "'"

   If Not .Adodc1.Recordset.EOF Then
   .Adodc1.Recordset.Find Criteri
   End If

   If .Adodc1.Recordset.EOF Then
   .Adodc1.Recordset.MoveFirst
   .Adodc1.Recordset.Find Criteri

      If .Adodc1.Recordset.EOF Then
      .Adodc1.Recordset.MoveFirst

      End If
   End If
End With


With EstructuraVis.ColBarra.Item(Str(BarraEnCurs))
```

```
    .Denominacio = frmTipusBarra.Adodc1.Recordset(1)
    .A = frmTipusBarra.Adodc1.Recordset(6) / 10000
    .Ix = frmTipusBarra.Adodc1.Recordset(8) / 100000000


    .Iy = frmTipusBarra.Adodc1.Recordset(11) / 100000000
    .Ip = frmTipusBarra.Adodc1.Recordset(16) / 100000000

    .h = frmTipusBarra.Adodc1.Recordset(2) / 1000
    .B = frmTipusBarra.Adodc1.Recordset(3) / 1000
    .DE = frmTipusBarra.Adodc1.Recordset(4) / 1000
    .DE1 = frmTipusBarra.Adodc1.Recordset(5) / 1000

End With



RedibuixaFormulariTipusBarra


End Sub

Sub ValidaCombo1()
Dim BarraEnCurs As Long
On Error Resume Next
BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs

    With frmTipusBarra
        Select Case .Combo1.Text

            Case "Sèrie IPN"
            .Adodc1.RecordSource = "select * from perfilsipn"
            EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil = 3
            EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Serie = "Sèrie IPN"

            Case "Sèrie IPE"
            .Adodc1.RecordSource = "select * from perfilsipe"
            EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil = 3
            EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Serie = "Sèrie IPE"

            Case "Sèrie HEB"
            .Adodc1.RecordSource = "select * from perfilsheb"
            EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil = 3
            EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Serie = "Sèrie HEB"

            Case "Sèrie HEA"
            .Adodc1.RecordSource = "select * from perfilshea"
            EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil = 3
            EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Serie = "Sèrie HEA"

            Case "Sèrie HEM"
```

```
            .Adodc1.RecordSource = "select * from perfilshem"
            EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil = 3
            EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Serie = "Sèrie HEM"

            Case "Sèrie UPN"
            .Adodc1.RecordSource = "select * from perfilsupn"
            EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil = 4
            EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Serie = "Sèrie UPN"

            Case "Sèrie L"
            .Adodc1.RecordSource = "select * from perfilsl"
            EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil = 5
            EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Serie = "Sèrie L"

            Case "Usuari"
            .Adodc1.RecordSource = "select * from perfilsdus"
            EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil = 9
            EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Serie = "Usuari"


        End Select


    .Adodc1.Refresh

    .DataCombo1.DataField = "Nom"
    .DataCombo1.Refresh
    .DataCombo1.DataField = ""

    End With



With EstructuraVis.ColBarra.Item(Str(BarraEnCurs))
    .Denominacio = frmTipusBarra.Adodc1.Recordset(1)
    .A = frmTipusBarra.Adodc1.Recordset(6) / 10000
    .Ix = frmTipusBarra.Adodc1.Recordset(8) / 100000000
    .Iy = frmTipusBarra.Adodc1.Recordset(11) / 100000000
    .Ip = frmTipusBarra.Adodc1.Recordset(16) / 100000000

    .h = frmTipusBarra.Adodc1.Recordset(2) / 1000
    .B = frmTipusBarra.Adodc1.Recordset(3) / 1000
    .DE = frmTipusBarra.Adodc1.Recordset(4) / 1000
    .DE1 = frmTipusBarra.Adodc1.Recordset(5) / 1000

End With

RedibuixaFormulariTipusBarra
```

```
End Sub


Sub ValidaDadesNouMaterial()
Dim E As Double, Ro As Double, Alfa As Double, Nu As Double, Nom As String
Dim Criteri As String
Dim BarraEnCurs As Long


BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs

With frmMaterialDefinitUsuari



   Select Case .Combo1.Text
   Case "MPa"
   E = Val(CDbl((.Text1.Text))) * 1000000#
   Case "Kp/cm^2"
    E = Val(CDbl((.Text1.Text))) * 100000
   End Select

   Nu = Val(CDbl((.Text2.Text)))
   Alfa = Val(CDbl((.Text3.Text)))

   Select Case .Combo2.Text
   Case "Kg/m^3"
   Ro = Val(CDbl(.Text4.Text))
   Case "g/cm^3"
   Ro = Val(CDbl(.Text4.Text)) * 1000
   End Select

   Nom = (.Text9.Text)
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Material = Nom

 End With


If AccesADades.ElNomDeMaterialJaExisteixONoEsValid(Nom) = False Then
'AFEGEIXO UN NOU REGISTRE A LA TAULA

   With frmTipusBarra.Adodc2.Recordset
```

```
    .AddNew
    !Nom = Nom
    !Densitat = Ro
    !E = E
    !Nu = Nu
    !Alfa = Alfa
    .Update

    End With
AccesADades.RedibuixaFormulariTipusBarra
frmTipusBarra.DataCombo2.Text = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Material
Unload frmMaterialDefinitUsuari
Else

End If

End Sub

Sub ValidaDadesNouPerfil()
Dim Nom As String, A As Double, Ix As Double, Iy As Double, Iz As Double
Dim Criteri As String
Dim BarraEnCurs As Long
Dim Auxiliar As String

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs

With frmSeccioGenerica
    Ix = Val(.Text1.Text) * 1
    Iy = Val(.Text2.Text) * 1
    Iz = Val(.Text3.Text) * 1
    A = Val(.Text4.Text) * 1
    'Ix = Val(.Text1.Text) * 0.00000001
    'Iy = Val(.Text2.Text) * 0.00000001
    'Iz = Val(.Text3.Text) * 0.00000001
    'A = Val(.Text4.Text) * 0.0001
    Nom = (.Text13.Text)

End With



If AccesADades.ElNomDePerfilJaExisteixONoEsValid(Nom) = False Then
'AFEGEIXO UN NOU REGISTRE A LA TAULA
    Auxiliar = frmTipusBarra.Adodc1.RecordSource
    frmTipusBarra.Adodc1.RecordSource = "select * from perfilsdus"
    With frmTipusBarra.Adodc1.Recordset

    .AddNew
    !Nom = Nom
    !Ix = Ix
```

```
    !Iy = Iy
    !Iz = Iz
    !A = A
    .Update

    End With
'AccesADades.RedibuixaFormulariTipusBarra
MsgBox "Secció agregada a la base de dades", vbInformation
'Unload frmSeccioGenerica
Else

End If


End Sub

Sub CarregaDadesNouMaterial()
Dim BarraEnCurs As Long
Dim E As Double, Ro As Double, Alfa As Double, Nu As Double

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs

With EstructuraVis.ColBarra.Item(Str(BarraEnCurs))
E = .E / 1000000
Ro = .Densitat
Alfa = .CoefDilatacioTermica
Nu = .Nu
End With


With frmMaterialDefinitUsuari

    Select Case .Combo1.Text
    Case "MPa"
    .Text1.Text = Format(E, "0.00E+00")
    Case "Kp/cm^2"
    .Text1.Text = Format(E * 10, "0.00E+00")
    End Select


    .Text2.Text = Format(Nu, "0.000")
    .Text3.Text = Format(Alfa, "0.00E-00")

    Select Case .Combo2.Text
    Case "Kg/m^3"
    .Text4.Text = Format(Ro, "0,000.0")
    Case "g/cm^3"
    .Text4.Text = Format(Ro / 1000, "00.000")
    End Select

End With
```

```
    With frmTipusBarra


    End With


End Sub


Sub ValidaToolbar1(Index As Integer)
Dim BarraEnCurs As Long
Dim Material As String
Dim Criteri As String

frmTipusBarra.Text7.SetFocus
BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs
Select Case Index
Case 1, 2
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil = Index
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).B = 0.3
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).h = 0.3

EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Material =
EstructuraVis.Preferencies.MaterialPerDefecteSeccionsNoMetaliques

AccesADades.RedibuixaFormulariTipusBarra
Case 3
AccesADades.DeterminaTipusPerfil
AccesADades.ValidaCombo1
AccesADades.RedibuixaFormulariTipusBarra
Case 4
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil = 9
frmSeccioGenerica.Show vbModal
Case 6
frmMaterialDefinitUsuari.Show vbModal
Case 7
Material = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Material

 If MsgBox("Esborrar el material '" & Material & "' de la base de dades?", vbYesNo) = vbYes
Then
    frmTipusBarra.Adodc2.Refresh
    With frmTipusBarra.Adodc2.Recordset
    '.Delete adAffectCurrent
    If .EOF = True Then .MoveFirst
    If .BOF = True Then .MoveFirst
    .Delete 1
    .Update
    frmTipusBarra.Adodc2.Refresh
    '.UpdateBatch
```

```
    '.MoveNext



    Select Case EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil
    Case 1, 2
    EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Material = EstructuraVis _
    .Preferencies.MaterialPerDefecteSeccionsNoMetaliques

    Case Is > 3
    EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Material = EstructuraVis _
    .Preferencies.MaterialPerDefecteSeccionsMetaliques

    End Select

    frmTipusBarra.DataCombo2.Text =
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Material

    AccesADades.RedibuixaFormulariTipusBarra
    End With


 End If


End Select
End Sub



Sub EscriuLesCaracteristiquesGeometriquesPerfilBaseDades()
Dim Cadena As String
Dim CasEnCurs As Integer
Dim i As Integer
Dim B As Double, h As Double, A As Double, Pes As Double, R As Double
Dim Ix As Double, Iy As Double, Wx As Double, Wy As Double
Dim Perimetre As Double, Material As String
Dim Densitat As Double, ModulYoung As Double, ModPoisson As Double, Alfa As Double
Dim Ip As Double, G As Double, Nom As String

Dim BarraEnCurs As Long, CoefDilatacioTermica As Double

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs


With EstructuraVis.ColBarra.Item(Str(BarraEnCurs))
CasEnCurs = .TipusPerfil
B = .B * 100
h = .h * 100
R = .R * 100
```

```
Select Case CasEnCurs
Case 1


.A = .B * .h
.Ix = .B * .h ^ 3 / 12
Wx = 1 / 12 * .B * .h ^ 3 / .h * 2
.Iy = 1 / 12 * .h * .B ^ 3
Wy = 1 / 12 * .h * .B ^ 3 / .B * 2
.Ip = .Ix + .Iy
.Denominacio = "REC" & B & "x" & h
.Serie = "NPCS"
Case 2
.A = Pi * .R ^ 2
.Ix = (Pi * (2 * .R) ^ 4) / 64
Wx = (Pi * (2 * .R) ^ 3) / 32
.Iy = (Pi * (2 * .R) ^ 4) / 64
Wy = (Pi * (2 * .R) ^ 3) / 32
.Denominacio = "C" & R
.Ip = .Ix + .Iy
.Serie = "NPCS"
Case 3

Wx = .Ix / .h * 2
Wy = .Iy / (.B) * 2

Case 4
Wx = .Ix / .h * 2
Wy = .Iy / (.B - .DE * 10)

Case 9
.Denominacio = "PDU"
.Serie = "Usuari"
End Select

Nom = .Denominacio
A = .A * 10000
Pes = .Densitat * .A
Ix = .Ix * 100000000
Iy = .Iy * 100000000
Ip = .Ip * 100000000
Wx = Wx * 1000000
Wy = Wy * 1000000
Alfa = .AngleOrientacio
ModulYoung = .E / 1000000
ModPoisson = .Nu

.G = .E / (2 * (1 + .Nu))
```

```
.GJ = .G * .Ip
G = .G / 1000000
Material = .Material
Densitat = .Densitat
CoefDilatacioTermica = .CoefDilatacioTermica
End With



'On Error GoTo Error
With frmTipusBarra

    Select Case CasEnCurs

    Case 1
    Cadena = "Perfil rectangular " & B & "x" & h & "cm" & vbCrLf & vbCrLf
    Cadena = Cadena & "Àrea: " & Format(A, "#,###.0") & " cm^2" & vbCrLf
    Cadena = Cadena & "Pes: " & Format(Pes, "#,###.0") & " Kg/m" & vbCrLf
    Cadena = Cadena & "Ix: " & Format(Ix, "#,###.00") & " cm^4" & vbCrLf
    Cadena = Cadena & "Wx: " & Format(Wx, "#,###.00") & " cm^3" & vbCrLf
    Cadena = Cadena & "Iy: " & Format(Iy, "#,###.00") & " cm^4" & vbCrLf
    Cadena = Cadena & "Wy: " & Format(Wy, "#,###.00") & " cm^3" & vbCrLf
    Cadena = Cadena & "Ip: " & Format(Ip, "#,###.00") & " cm^4" & vbCrLf
    Cadena = Cadena & "Perímetre: " & Format(2 * (B / 100) + 2 * (h / 100), "0.00") & "
m^2/m" & vbCrLf
    Cadena = Cadena & "Denominació: " & Nom & vbCrLf


    Case 2
    Cadena = "Perfil circular R=" & R & " cm" & vbCrLf & vbCrLf
    Cadena = Cadena & "Àrea: " & Format(A, "#,###.0") & " cm^2" & vbCrLf
    Cadena = Cadena & "Pes: " & Format(Pes, "#,###.0") & " Kg/m" & vbCrLf
    Cadena = Cadena & "Ix: " & Format(Ix, "#,###.00") & " cm^4" & vbCrLf
    Cadena = Cadena & "Wx: " & Format(Wx, "#,###.00") & " cm^3" & vbCrLf
    Cadena = Cadena & "Iy: " & Format(Iy, "#,###.00") & " cm^4" & vbCrLf
    Cadena = Cadena & "Wy: " & Format(Wy, "#,###.00") & " cm^3" & vbCrLf
    Cadena = Cadena & "Ip: " & Format(Ip, "#,###.00") & " cm^4" & vbCrLf
    Cadena = Cadena & "Perímetre: " & Format(2 * (B / 100) + 2 * (h / 100), "0.00") & "
m^2/m" & vbCrLf
    Cadena = Cadena & "Denominació: " & Nom & vbCrLf

    Case 3, 4, 5, 6, 7, 8
    Cadena = Cadena & Nom & vbCrLf & vbCrLf
    Cadena = Cadena & "Àrea: " & Format(A, "#,###.0") & " cm^2" & vbCrLf
    Cadena = Cadena & "Pes: " & Format(Pes, "#,###.0") & " Kg/m" & vbCrLf
    Cadena = Cadena & "Ix: " & Format(Ix, "#,###.00") & " cm^4" & vbCrLf
    Cadena = Cadena & "Wx: " & Format(Wx, "#,###.00") & " cm^3" & vbCrLf
    Cadena = Cadena & "Iy: " & Format(Iy, "#,###.00") & " cm^4" & vbCrLf
    Cadena = Cadena & "Wy: " & Format(Wy, "#,###.00") & " cm^3" & vbCrLf
    Cadena = Cadena & "Ip: " & Format(Ip, "#,###.000") & " cm^4" & vbCrLf
```

```
   'Cadena = Cadena & "Perímetre: " & Format(2 * (B / 100) + 2 * (h / 100), "0.00") & "
m^2/m" & vbCrLf


   Case 9
   Cadena = Cadena & "Perfil definit per l'usuari" & vbCrLf & vbCrLf
   Cadena = Cadena & "Àrea: " & Format(A, "#,###.00") & " cm^2" & vbCrLf
   Cadena = Cadena & "Pes: " & Format(Pes, "#,###.0") & " Kg/m" & vbCrLf
   Cadena = Cadena & "Ix: " & Format(Ix, "#,###.00") & " cm^4" & vbCrLf
   Cadena = Cadena & "Iy: " & Format(Iy, "#,###.00") & " cm^4" & vbCrLf
   Cadena = Cadena & "Ip: " & Format(Ip, "#,###.000") & " cm^4" & vbCrLf

   End Select

Cadena = Cadena & vbCrLf
Cadena = Cadena & "Material : " & Material & vbCrLf
Cadena = Cadena & "Densitat : " & Format(Densitat, "#,###.0") & " Kg/m^3" & vbCrLf
Cadena = Cadena & "Mod. Young : " & Format(ModulYoung, "0.0E+00") & "MPa" & vbCrLf
Cadena = Cadena & "Mod. Poisson : " & Format(ModPoisson, "0.000") & vbCrLf
Cadena = Cadena & "G : " & Format(G, "0.0E+00") & "Mpa" & vbCrLf
Cadena = Cadena & "Alfa : " & Format(CoefDilatacioTermica, "0.0E+00") & vbCrLf


.Text1.Text = Cadena

End With
'Exit Sub
'Error:

End Sub


Sub CarregaPreferenciesDefrmTipusBarra()
   With frmTipusBarra
   .Picture1.BackColor = EstructuraVis.Preferencies.ColorDeFons
   .Picture1.ScaleHeight = -140
   .Picture1.ScaleWidth = 140
   .Picture1.ScaleTop = 70
   .Picture1.ScaleLeft = -70
   .Text4.Visible = False
   .Text5.Visible = False
   .Text2.Visible = False
   .Text3.Visible = False
   End With
RedibuixaFormulariTipusBarra
Screen.MousePointer = vbDefault
End Sub

Private Sub DibuixaIPE()
Dim h As Double, B As Double, DE As Double, DE1 As Double
Dim Alfa As Double
```

```
On Error GoTo Error


    With frmTipusBarra

    If (.Adodc1.RecordSource <> "select * from perfilsipe") Then
    Exit Sub
    End If

    h = .Adodc1.Recordset(2)
    B = .Adodc1.Recordset(3)
    DE = .Adodc1.Recordset(4)
    DE1 = .Adodc1.Recordset(5)

    'Passem les dimensions a valors raonables perquè càpiguen a la finestra
    Alfa = h / 90
    h = 90
    B = B / Alfa
    DE = DE / Alfa
    DE1 = DE1 / Alfa

    .Picture1.Cls
    Call DibuixaEixosFTP
    .Picture1.DrawWidth = EstructuraVis.Preferencies.GruixDibuixIPES
    '.Picture1.TextWidth = EstructuraVis.Preferencies.TamanyTextIPES
    .Picture1.ForeColor = EstructuraVis.Preferencies.ColorDibuixIPES

    .Picture1.CurrentX = DE / 2
    .Picture1.CurrentY = 0
    .Picture1.Line Step(0, 0)-Step(0, h / 2 - DE1)
    .Picture1.Line Step(0, 0)-Step(B / 2 - DE / 2, 0)
    .Picture1.Line Step(0, 0)-Step(0, DE1)
    .Picture1.Line Step(0, 0)-Step(-B, 0)
    .Picture1.Line Step(0, 0)-Step(0, -DE1)
    .Picture1.Line Step(0, 0)-Step(B / 2 - DE / 2, 0)
    .Picture1.Line Step(0, 0)-Step(0, -h + 2 * DE1)
    .Picture1.Line Step(0, 0)-Step(-B / 2 + DE / 2, 0)
    .Picture1.Line Step(0, 0)-Step(0, -DE1)
    .Picture1.Line Step(0, 0)-Step(B, 0)
    .Picture1.Line Step(0, 0)-Step(0, DE1)
    .Picture1.Line Step(0, 0)-Step(-B / 2 + DE / 2, 0)
    .Picture1.Line Step(0, 0)-Step(0, h / 2 - DE1)

    .Picture1.ForeColor = EstructuraVis.Preferencies.ColorTextIPES
    .Picture1.Line (35, -45)-(35, 45)
    .Picture1.Line (35, -45)-(32, -40)
    .Picture1.Line (35, -45)-(38, -40)
    .Picture1.Line (35, 45)-(32, 40)
    .Picture1.Line (35, 45)-(38, 40)
    .Picture1.CurrentX = 35
    .Picture1.CurrentY = 0
```

```
    .Picture1.Print h * Alfa


    .Picture1.Line (-B / 2, 50)-(B / 2, 50)
    .Picture1.Line (-B / 2, 50)-(-B / 2 + 5, 53)
    .Picture1.Line (-B / 2, 50)-(-B / 2 + 5, 47)
    .Picture1.Line (B / 2, 50)-(B / 2 - 5, 53)
    .Picture1.Line (B / 2, 50)-(B / 2 - 5, 47)
    .Picture1.CurrentX = 0
    .Picture1.CurrentY = 61
    .Picture1.Print B * Alfa


    .Picture1.Line (-15, 0)-(15, 0)
    .Picture1.Line (DE / 2, 0)-(DE / 2 + 5, 3)
    .Picture1.Line (DE / 2, 0)-(DE / 2 + 5, -3)
    .Picture1.Line (-DE / 2, 0)-(-DE / 2 - 5, 3)
    .Picture1.Line (-DE / 2, 0)-(-DE / 2 - 5, -3)
    .Picture1.CurrentX = DE / 2 + 5
    .Picture1.CurrentY = -5
    .Picture1.Print DE * Alfa


    .Picture1.Line (-B * 2 / 6, -h / 2 + DE1 - 15)- _
    (-B * 2 / 6, -h / 2 + 15)
    .Picture1.Line (-B * 2 / 6 - 3, -h / 2 + DE1 + 5)- _
    (-B * 2 / 6, -h / 2 + DE1)
    .Picture1.Line (-B * 2 / 6 + 3, -h / 2 + DE1 + 5)- _
    (-B * 2 / 6, -h / 2 + DE1)
    .Picture1.Line (-B * 2 / 6 - 3, -h / 2 - 5)- _
    (-B * 2 / 6, -h / 2)
    .Picture1.Line (-B * 2 / 6, -h / 2)-(-B * 2 / 6 + 3, -h / 2 - 5)
    .Picture1.Print DE1 * Alfa

    .Picture1.CurrentX = -70
    .Picture1.CurrentY = 70
    .Picture1.Print "Cotes en mm"



    End With
Exit Sub
Error:

End Sub

Private Sub DibuixaHEB()
Dim h As Double, B As Double, DE As Double, DE1 As Double
Dim Alfa As Double
Dim BarraEnCurs As Long
```

```
On Error GoTo Error
BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs

    With frmTipusBarra

    'If (.Adodc1.RecordSource <> "select * from perfilsheb") Then
    'Exit Sub
    'End If
    If EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil <> 3 Then Exit Sub

    h = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).h * 1000
    B = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).B * 1000
    DE = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).DE * 1000
    DE1 = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).DE1 * 1000

    'Passem les dimensions a valors raonables perquè càpiguen a la finestra
    Alfa = h / 90
    h = 90
    B = B / Alfa
    DE = DE / Alfa
    DE1 = DE1 / Alfa

    .Picture1.Cls
    Call DibuixaEixosFTP

    .Picture1.ForeColor = EstructuraVis.Preferencies.ColorDibuixIPES
    .Picture1.DrawWidth = EstructuraVis.Preferencies.GruixDibuixIPES

    .Picture1.CurrentX = DE / 2
    .Picture1.CurrentY = 0
    .Picture1.Line Step(0, 0)-Step(0, h / 2 - DE1)
    .Picture1.Line Step(0, 0)-Step(B / 2 - DE / 2, 0)
    .Picture1.Line Step(0, 0)-Step(0, DE1)
    .Picture1.Line Step(0, 0)-Step(-B, 0)
    .Picture1.Line Step(0, 0)-Step(0, -DE1)
    .Picture1.Line Step(0, 0)-Step(B / 2 - DE / 2, 0)
    .Picture1.Line Step(0, 0)-Step(0, -h + 2 * DE1)
    .Picture1.Line Step(0, 0)-Step(-B / 2 + DE / 2, 0)
    .Picture1.Line Step(0, 0)-Step(0, -DE1)
    .Picture1.Line Step(0, 0)-Step(B, 0)
    .Picture1.Line Step(0, 0)-Step(0, DE1)
    .Picture1.Line Step(0, 0)-Step(-B / 2 + DE / 2, 0)
    .Picture1.Line Step(0, 0)-Step(0, h / 2 - DE1)

    .Picture1.ForeColor = EstructuraVis.Preferencies.ColorTextIPES
    .Picture1.Line (35, -45)-(35, 45)
    .Picture1.Line (35, -45)-(32, -40)
    .Picture1.Line (35, -45)-(38, -40)
    .Picture1.Line (35, 45)-(32, 40)
    .Picture1.Line (35, 45)-(38, 40)
    .Picture1.CurrentX = 35
```

```
    .Picture1.CurrentY = 0
    .Picture1.Print h * Alfa


    .Picture1.Line (-B / 2, 50)-(B / 2, 50)
    .Picture1.Line (-B / 2, 50)-(-B / 2 + 5, 53)
    .Picture1.Line (-B / 2, 50)-(-B / 2 + 5, 47)
    .Picture1.Line (B / 2, 50)-(B / 2 - 5, 53)
    .Picture1.Line (B / 2, 50)-(B / 2 - 5, 47)
    .Picture1.CurrentX = 0
    .Picture1.CurrentY = 61
    .Picture1.Print B * Alfa


    .Picture1.Line (-15, 0)-(15, 0)
    .Picture1.Line (DE / 2, 0)-(DE / 2 + 5, 3)
    .Picture1.Line (DE / 2, 0)-(DE / 2 + 5, -3)
    .Picture1.Line (-DE / 2, 0)-(-DE / 2 - 5, 3)
    .Picture1.Line (-DE / 2, 0)-(-DE / 2 - 5, -3)
    .Picture1.CurrentX = DE / 2 + 5
    .Picture1.CurrentY = -5
    .Picture1.Print DE * Alfa


    .Picture1.Line (-B * 2 / 6, -h / 2 + DE1 - 15)- _
    (-B * 2 / 6, -h / 2 + 15)
    .Picture1.Line (-B * 2 / 6 - 3, -h / 2 + DE1 + 5)- _
    (-B * 2 / 6, -h / 2 + DE1)
    .Picture1.Line (-B * 2 / 6 + 3, -h / 2 + DE1 + 5)- _
    (-B * 2 / 6, -h / 2 + DE1)
    .Picture1.Line (-B * 2 / 6 - 3, -h / 2 - 5)- _
    (-B * 2 / 6, -h / 2)
    .Picture1.Line (-B * 2 / 6, -h / 2)-(-B * 2 / 6 + 3, -h / 2 - 5)
    .Picture1.Print DE1 * Alfa

    .Picture1.CurrentX = -70
    .Picture1.CurrentY = 70
    .Picture1.Print "Cotes en mm"



    End With
Exit Sub
Error:

End Sub


Private Sub DibuixaL()
Dim h As Double, B As Double, DE As Double, DE1 As Double
Dim Alfa As Double
```

```
Dim BarraEnCurs As Long

'On Error GoTo Error
BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs

   With frmTipusBarra


   If EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil <> 5 Then Exit Sub

   B = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).B * 1000
   h = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).h * 1000 * 10
   DE = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).DE * 1000
   DE1 = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).DE1 * 1000

   'REDUEIXO LES DIMENSIONS PERQUÈ CÀPIGUEN A LA PANTALLA
   Alfa = B / 90
   B = 90
   h = h / Alfa
   DE = DE / Alfa
   DE1 = DE1 / Alfa

   .Picture1.Cls
   Call DibuixaEixosFTP

   .Picture1.ForeColor = EstructuraVis.Preferencies.ColorDibuixIPES
   .Picture1.DrawWidth = EstructuraVis.Preferencies.GruixDibuixIPES

   .Picture1.CurrentX = -B + h
   .Picture1.CurrentY = -h + DE
   .Picture1.Line Step(0, 0)-Step(B - DE, 0)
   .Picture1.Line Step(0, 0)-Step(0, B - DE)
   .Picture1.Line Step(0, 0)-Step(DE, 0)
   .Picture1.Line Step(0, 0)-Step(0, -B)
   .Picture1.Line Step(0, 0)-Step(-B, 0)
   .Picture1.Line Step(0, 0)-Step(0, DE)


   .Picture1.ForeColor = EstructuraVis.Preferencies.ColorTextIPES
   .Picture1.Line (40, B - h)-(40, -h)
   .Picture1.Line (40, B - h)-Step(-3, -5)
   .Picture1.Line (40, B - h)-Step(3, -5)
   .Picture1.Line (40, -h)-Step(-3, 5)
   .Picture1.Line (40, -h)-Step(3, 5)

   .Picture1.CurrentX = 45
   .Picture1.CurrentY = (B - h) / 2
   .Picture1.Print (B) * Alfa
```

```
    .Picture1.Line (-20, -h + DE + 10)-(-20, -h - 10)
    .Picture1.Line (-20, -h + DE)-Step(-3, 5)
    .Picture1.Line (-20, -h + DE)-Step(3, 5)
    .Picture1.Line (-20, -h)-Step(-3, -5)
    .Picture1.Line (-20, -h)-Step(3, -5)

    .Picture1.CurrentX = -20
    .Picture1.CurrentY = -h - 15
    .Picture1.Print DE * Alfa



    .Picture1.CurrentX = -70
    .Picture1.CurrentY = 70
    .Picture1.Print "Cotes en mm"



    End With
Exit Sub
Error:

End Sub

Private Sub DibuixaUPN()
Dim h As Double, B As Double, DE As Double, DE1 As Double
Dim Alfa As Double
Dim BarraEnCurs As Long

'On Error GoTo Error
BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs

    With frmTipusBarra


    If EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil <> 4 Then Exit Sub

    h = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).h * 1000
    B = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).B * 1000
    DE = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).DE * 1000 * 10 '(ÉS EN CM)
    DE1 = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).DE1 * 1000

    'REDUEIXO LES DIMENSIONS PERQUÈ CÀPIGUEN A LA PANTALLA
    Alfa = h / 90
    h = 90
    B = B / Alfa
    DE = DE / Alfa
    DE1 = DE1 / Alfa

    .Picture1.Cls
```

```
Call DibuixaEixosFTP

.Picture1.ForeColor = EstructuraVis.Preferencies.ColorDibuixIPES
.Picture1.DrawWidth = EstructuraVis.Preferencies.GruixDibuixIPES

.Picture1.CurrentY = h / 2
.Picture1.CurrentX = -DE
.Picture1.Line Step(0, 0)-Step(B, 0)
.Picture1.Line Step(0, 0)-Step(0, -DE1)
.Picture1.Line Step(0, 0)-Step(DE1 - B, 0)
.Picture1.Line Step(0, 0)-Step(0, -h + DE1 * 2)
.Picture1.Line Step(0, 0)-Step(B - DE1, 0)
.Picture1.Line Step(0, 0)-Step(0, -DE1)
.Picture1.Line Step(0, 0)-Step(-B, 0)
.Picture1.Line Step(0, 0)-Step(0, h)


.Picture1.ForeColor = EstructuraVis.Preferencies.ColorTextIPES
.Picture1.Line (40, -45)-(40, 45)
.Picture1.Line (40, -45)-(37, -40)
.Picture1.Line (40, -45)-(43, -40)
.Picture1.Line (40, 45)-(37, 40)
.Picture1.Line (40, 45)-(43, 40)
.Picture1.CurrentX = 45
.Picture1.CurrentY = 0
.Picture1.Print h * Alfa


.Picture1.Line ((-DE), 50)-(B - DE, 50)
.Picture1.Line (-DE, 50)-(-DE + 5, 53)
.Picture1.Line (-DE, 50)-(-DE + 5, 47)
.Picture1.Line ((B - DE), 50)-((B) - DE - 5, 53)
.Picture1.Line ((B - DE), 50)-((B) - DE - 5, 47)
.Picture1.CurrentX = (B - DE) / 3
.Picture1.CurrentY = 61
.Picture1.Print B * Alfa


'.Picture1.Line (-15, 0)-(15, 0)
'.Picture1.Line (DE / 2, 0)-(DE / 2 + 5, 3)
'.Picture1.Line (DE / 2, 0)-(DE / 2 + 5, -3)
'.Picture1.Line (-DE / 2, 0)-(-DE / 2 - 5, 3)
'.Picture1.Line (-DE / 2, 0)-(-DE / 2 - 5, -3)
'.Picture1.CurrentX = DE / 2 + 5
'.Picture1.CurrentY = -5
'.Picture1.Print DE * Alfa


'.Picture1.Line (-B * 2 / 6, -h / 2 + DE1 - 15)- _
(-B * 2 / 6, -h / 2 + 15)
'.Picture1.Line (-B * 2 / 6 - 3, -h / 2 + DE1 + 5)- _
```

```
    (-B * 2 / 6, -h / 2 + DE1)
    '.Picture1.Line (-B * 2 / 6 + 3, -h / 2 + DE1 + 5)- _
    (-B * 2 / 6, -h / 2 + DE1)
    '.Picture1.Line (-B * 2 / 6 - 3, -h / 2 - 5)- _
    (-B * 2 / 6, -h / 2)
    '.Picture1.Line (-B * 2 / 6, -h / 2)-(-B * 2 / 6 + 3, -h / 2 - 5)
    '.Picture1.Print DE1 * Alfa

    .Picture1.CurrentX = -70
    .Picture1.CurrentY = 70
    .Picture1.Print "Cotes en mm"



    End With
Exit Sub
Error:

End Sub


Private Sub DibuixaEixosFTP()
    With frmTipusBarra
    .Picture1.DrawWidth = 1
    .Picture1.AutoRedraw = True
    .Picture1.DrawWidth = EstructuraVis.Preferencies.GruixEixosCoordenats
    .Picture1.ForeColor = EstructuraVis.Preferencies.ColorEixXFormularis
    .Picture1.Line (-70, 0)-(70, 0)
    .Picture1.DrawWidth = EstructuraVis.Preferencies.GruixEixosCoordenats
    .Picture1.ForeColor = EstructuraVis.Preferencies.ColorEixYFormularis
    .Picture1.Line (0, 70)-(0, -70)

    End With
End Sub


Private Sub DibuixaPerfilRectangular()
Dim Alfa As Double
Dim Hf As Double, Bf As Double
Dim BarraEnCurs As Long

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs

With EstructuraVis.ColBarra.Item(Str(BarraEnCurs))
If .h = 0 Then .h = 0.3
If .B = 0 Then .B = 0.3

Hf = .h * 100
Bf = .B * 100
Alfa = .AngleOrientacio
```

```
End With

With frmTipusBarra

    .Picture1.Cls

    'Passem les dimensions a valors raonables perquè càpiguen a la finestra
    If Hf = 0 Then
    Hf = 30
    EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).h = 0.3
    End If

    If Bf = 0 Then
    EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).B = 0.3
    Bf = 30
    End If

    DibuixaEixosFTP
    If Hf > Bf Then
    Alfa = Hf / 90
    Hf = 90
    Bf = Bf / Alfa
    Else
    Alfa = Bf / 90
    Bf = 90
    Hf = Hf / Alfa
    End If

    .Picture1.CurrentX = Bf / 2
    .Picture1.CurrentY = -Hf / 2
    .Picture1.ForeColor = EstructuraVis.Preferencies.ColorDibuixIPES
    .Picture1.DrawWidth = EstructuraVis.Preferencies.GruixDibuixIPES
    .Picture1.Line Step(0, 0)-Step(0, Hf)
    .Picture1.Line Step(0, 0)-Step(-Bf, 0)
    .Picture1.Line Step(0, 0)-Step(0, -Hf)
    .Picture1.Line Step(0, 0)-Step(Bf, 0)
    .Picture1.ForeColor = EstructuraVis.Preferencies.ColorTextIPES


    .Picture1.CurrentX = 0
    .Picture1.CurrentY = -Hf / 2
    .Picture1.Print Bf * Alfa
    .Picture1.Line (-Bf / 2, -Hf / 2 - 8)-(Bf / 2, -Hf / 2 - 8)
    .Picture1.Line (-Bf / 2, -Hf / 2 - 8)-Step(5, 3)
    .Picture1.Line (-Bf / 2, -Hf / 2 - 8)-Step(5, -3)
    .Picture1.Line (Bf / 2, -Hf / 2 - 8)-Step(-5, 3)
    .Picture1.Line (Bf / 2, -Hf / 2 - 8)-Step(-5, -3)
```

```
    .Picture1.CurrentX = Bf / 2 + 8
    .Picture1.CurrentY = 0
    .Picture1.Print (Hf * Alfa)
    .Picture1.Line (Bf / 2 + 8, -Hf / 2)-(Bf / 2 + 8, Hf / 2)
    .Picture1.Line (Bf / 2 + 8, -Hf / 2)-Step(3, 5)
    .Picture1.Line (Bf / 2 + 8, -Hf / 2)-Step(-3, 5)
    .Picture1.Line (Bf / 2 + 8, Hf / 2)-Step(3, -5)
    .Picture1.Line (Bf / 2 + 8, Hf / 2)-Step(-3, -5)


    .Picture1.CurrentX = -70
    .Picture1.CurrentY = 70
    .Picture1.Print "Cotes en cm"

End With

End Sub

'
'
Sub RedibuixaFormulariTipusBarra()
Dim CasEnCurs As Double
Dim i As Integer
Dim BarraEnCurs As Long


BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs
CasEnCurs = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil

    With frmTipusBarra

    If CasEnCurs = 0 Then
    EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil = 1
    CasEnCurs = 1
    End If

    For i = 1 To 3
    .Toolbar1.Buttons.Item(i).Value = tbrUnpressed
    Next i

        Select Case CasEnCurs

        Case 1
        .Toolbar1.Buttons.Item(1).Value = tbrPressed
        Case 2
        .Toolbar1.Buttons.Item(2).Value = tbrPressed
        Case 3, 4, 5, 6, 7, 8, 9
        .Toolbar1.Buttons.Item(3).Value = tbrPressed
        Case 10
        .Toolbar1.Buttons.Item(4).Value = tbrPressed
        End Select
```

```
AssignaDadesMaterial

Select Case CasEnCurs
Case 1

.Combo1.Visible = False
.DataCombo1.Visible = False
DibuixaPerfilRectangular

Case 2

.Combo1.Visible = False
.DataCombo1.Visible = False
DibuixaPerfilRodo

Case 3
AccesADades.AssignaDadesPerfilMetalic
.Combo1.Visible = True
.DataCombo1.Visible = True
DibuixaHEB

Case 4
AccesADades.AssignaDadesPerfilMetalic
.Combo1.Visible = True
.DataCombo1.Visible = True
DibuixaUPN
Case 5
AccesADades.AssignaDadesPerfilMetalic
.Combo1.Visible = True
.DataCombo1.Visible = True
DibuixaL
Case 6

Case 7

Case 8


Case 9
AccesADades.AssignaDadesPerfilMetalic
.Combo1.Visible = True
.DataCombo1.Visible = True
DibuixaDadesPerfilGeneric
End Select

.DataCombo2.Text = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Material

.Check1.Caption = "Node " & EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).NumNode1
& " articulat"
```

```
   .Check2.Caption = "Node " & EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).NumNode2
& " articulat"

   If EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Node1Articulat = True Then
   .Check1.Value = 1
   Else
   .Check1.Value = 0
   End If

   If EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Node2Articulat = True Then
   .Check2.Value = 1
   Else
   .Check2.Value = 0
   End If

   End With

EscriuLesCaracteristiquesGeometriquesPerfilBaseDades
DibuixaFinestreta3FTP
frmTipusBarra.Command1.Caption = "Acceptar"


'If EstructuraVis.AuxFormularis.EditarBarraEnPantalla = True Then
'frmTipusBarra.Command2.Enabled = False
'Else
'frmTipusBarra.Command2.Enabled = True
'End If

frmTipusBarra.Tag = 0
End Sub


Sub CanviEnToolBar1FTP(Index As Integer)
Dim BarraEnCurs As Long, CasEnCurs As Integer

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil = Index
Select Case Index
Case 1, 2
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Material =
EstructuraVis.Preferencies.MaterialPerDefecteSeccionsNoMetaliques
Case 3
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Material =
EstructuraVis.Preferencies.MaterialPerDefecteSeccionsMetaliques
End Select


RedibuixaFormulariTipusBarra
End Sub

Private Sub DibuixaPerfilRodo()
```

```
Dim rf As Double
Dim BarraEnCurs As Long

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs

With EstructuraVis.ColBarra.Item(Str(BarraEnCurs))
If .R = 0 Then .R = 0.15
rf = 45
End With


With frmTipusBarra

.Picture1.Cls
DibuixaEixosFTP
.Picture1.DrawWidth = EstructuraVis.Preferencies.GruixDibuixIPES
.Picture1.ForeColor = EstructuraVis.Preferencies.ColorDibuixIPES
.Picture1.Circle (0, 0), 45
.Picture1.ForeColor = EstructuraVis.Preferencies.ColorTextIPES
.Picture1.Line (0, 0)-(0, -45)
.Picture1.Line (0, -45)-Step(-3, 5)
.Picture1.Line (0, -45)-Step(3, 5)
.Picture1.CurrentX = 0
.Picture1.CurrentY = -rf / 2
.Picture1.Print EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).R * 100
.Picture1.CurrentX = -70
.Picture1.CurrentY = 70
.Picture1.Print "Cota en cm"

End With
End Sub

Sub ValidaCombo3()
Dim Criteri As String
Dim BarraEnCurs As Long
Dim E As Double
Dim Nu As Double
Dim Material As String
Dim Densitat As Double
Dim CoefDilatacioTermica As Double

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs

   With EstructuraVis.ColBarra.Item(Str(BarraEnCurs))
   .Material = frmTipusBarra.DataCombo2.Text
   End With



   With frmTipusBarra
   '.DataCombo2.ReFill
```

```
    .DataCombo2.DataField = ""
    .Adodc2.Recordset.MoveNext
     Criteri = "Nom LIKE " & "'" & .DataCombo2.Text & "'"

    If Not .Adodc2.Recordset.EOF Then
    .Adodc2.Recordset.Find Criteri
    End If

    If .Adodc2.Recordset.EOF Then
    .Adodc2.Recordset.MoveFirst
    .Adodc2.Recordset.Find Criteri

       If .Adodc2.Recordset.EOF Then
       .Adodc2.Recordset.MoveFirst

       End If
    End If
    .DataCombo2.DataField = ""
    .DataCombo2.Refresh

    E = .Adodc2.Recordset(3)
    Nu = .Adodc2.Recordset(4)
    Material = .Adodc2.Recordset(1)
    Densitat = .Adodc2.Recordset(2)
    CoefDilatacioTermica = .Adodc2.Recordset(5)

    End With

    With EstructuraVis.ColBarra.Item(BarraEnCurs)
    .E = E
    .Nu = Nu
    .Material = Material
    .Densitat = Densitat
    .CoefDilatacioTermica = CoefDilatacioTermica
    End With

RedibuixaFormulariTipusBarra


End Sub


Sub ClickEnPicture1FTP(X As Double, Y As Double)
Dim CasEnCurs As Double
Dim i As Integer
Dim Bf As Double
Dim Hf As Double
Dim Alfa As Double
Dim rf As Double
Dim BarraEnCurs As Long
```

```
BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs
   With frmTipusBarra

   For i = 1 To 3
   If .Toolbar1.Buttons.Item(i).Value = tbrPressed Then CasEnCurs = i
   Next i

   With EstructuraVis.ColBarra.Item(Str(BarraEnCurs))
   Hf = .h * 100
   Bf = .B * 100
   'rf = .R * 100
   rf = 45
   If Hf > Bf Then
   Alfa = Hf / 90
   Hf = 90
   Bf = Bf / Alfa
   Else
   Alfa = Bf / 90
   Bf = 90
   Hf = Hf / Alfa
   End If
   End With


   Select Case EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil
   Case 1
      If (((X) ^ 2 + (Y + Hf / 2) ^ 2) < 250) Then
      'Canviem el valor de B
      .Text5.Left = 0
      .Text5.Top = -Hf / 2
      .Text5.Visible = True
      .Text5.Text = ""
      .Text5.SetFocus
      End If

      If (((X - Bf / 2 - 8) ^ 2 + (Y) ^ 2) < 250) Then
      'Canviem el valor de h
      .Text4.Top = 0
      .Text4.Left = Bf / 2 + 8
      .Text4.Visible = True
      .Text4.Text = ""
      .Text4.SetFocus
      End If
   Case 2
      If (((X) ^ 2 + (Y + rf / 2) ^ 2) < 250) Then
      'Canviem el valor de h
      .Text2.Left = 0
      .Text2.Top = -rf / 2
      .Text2.Visible = True
      .Text2.Text = ""
```

```
        .Text2.SetFocus
        End If

    End Select

    End With
End Sub

Sub ClickEnPicture2FTP(X As Double, Y As Double)


    With frmTipusBarra

        .Text3.Visible = True
        .Text3.Text = ""
        .Text3.SetFocus

    End With
End Sub


Sub PerdEnfocText5FTP()
Dim BarraEnCurs As Long

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs
With frmTipusBarra
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).B = _
Val(.Text5.Text) / 100
.Text5.Visible = False
DibuixaPerfilRectangular
End With
AccesADades.RedibuixaFormulariTipusBarra
End Sub

Sub PerdEnfocText2FTP()
Dim BarraEnCurs As Long
BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs
With frmTipusBarra
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).R = _
 (Val(.Text2.Text) / 100)
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Serie = _
 "NPCS"
.Text2.Visible = False
DibuixaPerfilRodo
End With
AccesADades.RedibuixaFormulariTipusBarra
End Sub

Sub PerdEnfocText3FTP()
Dim BarraEnCurs As Long
```

```
BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs
With frmTipusBarra
EstructuraVis.AuxFormularis.AngleOrientacio = Val(.Text3.Text) * Pi / 180
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).AngleOrientacio = _
Val(.Text3.Text) * Pi / 180

.Text3.Visible = False
DibuixaFinestreta3FTP
End With
RedibuixaFormulariTipusBarra
End Sub

Sub PerdEnfocText4FTP()
Dim BarraEnCurs As Long

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs
With frmTipusBarra
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).h = _
Val(.Text4.Text) / 100

'HQ = Val(.Text4.Text)
.Text4.Visible = False

DibuixaPerfilRectangular
End With
AccesADades.RedibuixaFormulariTipusBarra
End Sub


Private Sub DibuixaFinestreta3FTP()

Dim X1 As Double, Y1 As Double
Dim X2 As Double, Y2 As Double
Dim X3 As Double, Y3 As Double
Dim X4 As Double, Y4 As Double
Dim Alfa As Double
Dim BarraEnCurs As Long

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs
Alfa = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).AngleOrientacio

Select Case (EstructuraVis.ColBarra.Item(Str( _
EstructuraVis.AuxFormularis.BarraEnCurs)).EsUnPilar)

Case False

   With frmTipusBarra
   .Picture2.AutoRedraw = True
   .Picture2.Cls
   .Picture2.ScaleMode = 0
   .Picture2.ScaleLeft = -50
```

```
.Picture2.ScaleWidth = 100
.Picture2.ScaleTop = 50
.Picture2.ScaleHeight = -100

'Aixo es provisional
.Picture2.ForeColor = EstructuraVis.Preferencies.ColorTextAnglesAlfaIBeta
.Picture2.DrawStyle = 0
.Picture2.DrawWidth = EstructuraVis.Preferencies.GruixEixosCoordenats
.Picture2.FontSize = EstructuraVis.Preferencies.GrandariaTextEixosCoordenats
.Picture2.BackColor = EstructuraVis.Preferencies.ColorDeFons
.Picture2.Line (-50, 0)-(50, 0), EstructuraVis.Preferencies.ColorTextAnglesAlfaIBeta
.Picture2.CurrentX = 43
.Picture2.CurrentY = 0
.Picture2.Print "xz"
.Picture2.Line (0, 0)-(0, 50), EstructuraVis.Preferencies.ColorEixY
.Picture2.CurrentX = 0
.Picture2.CurrentY = 50
.Picture2.Print "y"

.Picture2.DrawWidth = EstructuraVis.Preferencies.GruixEixosLocals

'Dibuixo l'eix local y
.Picture2.Line (0, 0)-(-30 * Sin(Alfa), _
30 * Cos(Alfa)), _
EstructuraVis.Preferencies.ColorEixY

.Picture2.CurrentX = 30 * Cos(Alfa + Pi / 2)
.Picture2.CurrentY = 30 * Sin(Alfa + Pi / 2)

.Picture2.Line Step(0, 0)- _
Step(-9 * Cos(Pi / 2 + Alfa + Pi / 5), -9 * Sin(Pi / 2 + Alfa + Pi / 5)), _
EstructuraVis.Preferencies.ColorEixY
.Picture2.Print "y local"
.Picture2.CurrentX = 30 * Cos(Alfa + Pi / 2)
.Picture2.CurrentY = 30 * Sin(Alfa + Pi / 2)

.Picture2.Line Step(0, 0)- _
Step(9 * Cos(-Alfa + Pi / 5 + Pi / 2), -9 * Sin(-Alfa + Pi / 5 + Pi / 2)), _
EstructuraVis.Preferencies.ColorEixY

'Ara dibuixo l'eix local x
.Picture2.Line (0, 0)-(-30 * Cos(Alfa), _
-30 * Sin(Alfa)), _
EstructuraVis.Preferencies.ColorEixX

.Picture2.CurrentX = -30 * Cos(Alfa)
.Picture2.CurrentY = -30 * Sin(Alfa)

.Picture2.Line Step(0, 0)- _
Step(9 * Cos(Alfa + Pi / 5), 9 * Sin(Alfa + Pi / 5)), _
EstructuraVis.Preferencies.ColorEixX
```

```
    .Picture2.CurrentX = -30 * Cos(Alfa)
    .Picture2.CurrentY = -30 * Sin(Alfa)

    .Picture2.Line Step(0, 0)- _
    Step(9 * Cos(-Alfa + Pi / 5), -9 * Sin(-Alfa + Pi / 5)), _
    EstructuraVis.Preferencies.ColorEixX
    .Picture2.Print "x local"

    .Picture2.CurrentX = -45
    .Picture2.CurrentY = 45
    .Picture2.ForeColor = EstructuraVis.Preferencies.ColorTextAnglesAlfaIBeta
    .Picture2.Print "Alfa=" & Format(Alfa * 180 / Pi, "##0.0")


    'Ara dibuixem la biga
    .Picture2.DrawWidth = EstructuraVis.Preferencies.GruixBigaOrientacio
    X1 = 40 * Cos(Alfa + Pi / 6 + Pi / 2)
    Y1 = 40 * Sin(Alfa + Pi / 6 + Pi / 2)
    X2 = 40 * Cos(Alfa - Pi / 6 + Pi / 2)
    Y2 = 40 * Sin(Alfa - Pi / 6 + Pi / 2)
    X3 = 40 * Cos(Alfa + Pi / 6 + Pi * 3 / 2)
    Y3 = 40 * Sin(Alfa + Pi / 6 + Pi * 3 / 2)
    X4 = 40 * Cos(Alfa - Pi / 6 + Pi * 3 / 2)
    Y4 = 40 * Sin(Alfa - Pi / 6 + Pi * 3 / 2)


    .Picture2.Line (X1, Y1)-(X2, Y2), EstructuraVis.Preferencies.ColorBigaOrientacio
    .Picture2.Line (X3, Y3)-(X2, Y2), EstructuraVis.Preferencies.ColorBigaOrientacio
    .Picture2.Line (X3, Y3)-(X4, Y4), EstructuraVis.Preferencies.ColorBigaOrientacio
    .Picture2.Line (X1, Y1)-(X4, Y4), EstructuraVis.Preferencies.ColorBigaOrientacio

    End With

Case True

    With frmTipusBarra
    .Picture2.AutoRedraw = True
    .Picture2.Cls
    .Picture2.ScaleMode = 0
    .Picture2.ScaleLeft = -50
    .Picture2.ScaleWidth = 100
    .Picture2.ScaleTop = 50
    .Picture2.ScaleHeight = -100

    'Aixo es provisional
    .Picture2.ForeColor = EstructuraVis.Preferencies.ColorTextAnglesAlfaIBeta
    .Picture2.DrawStyle = 0
    .Picture2.DrawWidth = EstructuraVis.Preferencies.GruixEixosCoordenats
    .Picture2.FontSize = EstructuraVis.Preferencies.GrandariaTextEixosCoordenats
    .Picture2.BackColor = EstructuraVis.Preferencies.ColorDeFons
```

```
.Picture2.Line (0, 0)-(50, 0), EstructuraVis.Preferencies.ColorEixX
.Picture2.CurrentX = 43
.Picture2.CurrentY = 0
.Picture2.Print "x"
.Picture2.Line (0, 0)-(0, -50), EstructuraVis.Preferencies.ColorEixZ
.Picture2.CurrentX = 0
.Picture2.CurrentY = -40
.Picture2.Print "z"

.Picture2.DrawWidth = EstructuraVis.Preferencies.GruixEixosLocals

'Dibuixo l'eix local x
.Picture2.Line (0, 0)-(-30 * Sin(Alfa), _
30 * Cos(Alfa)), _
EstructuraVis.Preferencies.ColorEixX

.Picture2.CurrentX = -30 * Sin(Alfa)
.Picture2.CurrentY = 30 * Cos(Alfa)

.Picture2.Line Step(0, 0)- _
Step(-9 * Cos(Pi / 2 + Alfa + Pi / 5), -9 * Sin(Pi / 2 + Alfa + Pi / 5)), _
EstructuraVis.Preferencies.ColorEixX
.Picture2.Print "x local"
.Picture2.CurrentX = -30 * Sin(Alfa)
.Picture2.CurrentY = 30 * Cos(Alfa)

.Picture2.Line Step(0, 0)- _
Step(9 * Cos(-Alfa + Pi / 5 + Pi / 2), -9 * Sin(-Alfa + Pi / 5 + Pi / 2)), _
EstructuraVis.Preferencies.ColorEixX

'Ara dibuixo l'eix local y
.Picture2.Line (0, 0)-(-30 * Cos(Alfa), _
-30 * Sin(Alfa)), _
EstructuraVis.Preferencies.ColorEixY

.Picture2.CurrentX = -30 * Cos(Alfa)
.Picture2.CurrentY = -30 * Sin(Alfa)

.Picture2.Line Step(0, 0)- _
Step(9 * Cos(Alfa + Pi / 5), 9 * Sin(Alfa + Pi / 5)), _
EstructuraVis.Preferencies.ColorEixY

.Picture2.CurrentX = -30 * Cos(Alfa)
.Picture2.CurrentY = -30 * Sin(Alfa)

.Picture2.Line Step(0, 0)- _
Step(9 * Cos(-Alfa + Pi / 5), -9 * Sin(-Alfa + Pi / 5)), _
EstructuraVis.Preferencies.ColorEixY
.Picture2.Print "y local"

.Picture2.CurrentX = -45
```

```
    .Picture2.CurrentY = 45
    .Picture2.ForeColor = EstructuraVis.Preferencies.ColorTextAnglesAlfaIBeta
    .Picture2.Print "Alfa=" & Format(Alfa * 180 / Pi, "##0.0")


    'Ara dibuixem la biga
    .Picture2.DrawWidth = EstructuraVis.Preferencies.GruixBigaOrientacio
    X1 = 40 * Cos(Alfa + Pi / 6 + Pi / 2 - Pi / 2)
    Y1 = 40 * Sin(Alfa + Pi / 6 + Pi / 2 - Pi / 2)
    X2 = 40 * Cos(Alfa - Pi / 6 + Pi / 2 - Pi / 2)
    Y2 = 40 * Sin(Alfa - Pi / 6 + Pi / 2 - Pi / 2)
    X3 = 40 * Cos(Alfa + Pi / 6 + Pi * 3 / 2 - Pi / 2)
    Y3 = 40 * Sin(Alfa + Pi / 6 + Pi * 3 / 2 - Pi / 2)
    X4 = 40 * Cos(Alfa - Pi / 6 + Pi * 3 / 2 - Pi / 2)
    Y4 = 40 * Sin(Alfa - Pi / 6 + Pi * 3 / 2 - Pi / 2)


    .Picture2.Line (X1, Y1)-(X2, Y2), EstructuraVis.Preferencies.ColorBigaOrientacio
    .Picture2.Line (X3, Y3)-(X2, Y2), EstructuraVis.Preferencies.ColorBigaOrientacio
    .Picture2.Line (X3, Y3)-(X4, Y4), EstructuraVis.Preferencies.ColorBigaOrientacio
    .Picture2.Line (X1, Y1)-(X4, Y4), EstructuraVis.Preferencies.ColorBigaOrientacio

    End With


End Select


End Sub



Sub CapturaDadesPerfilCircular()

    'R=R
    'A = Pi * R ^ 2
    'Ix = Pi / 64 * (2 * R) ^ 4
    'Iy = Pi / 64 * (2 * R) ^ 4

    'With frmTipusBarra
    'Densitat = .Adodc2.Recordset(2)
    'Nom = "CIRCULAR R=" & R
    'E = .Adodc2.Recordset(3) * 10000 * GGrav
    'Material = .Adodc2.Recordset(1)
    'End With

    'B = 0
    'h = 0
    'DE = 0
    'DE1 = 0
```

```
    'TipusPerfil = 2

End Sub

Sub CapturaDadesPerfilIP()

    'With frmTipusBarra

    'Nom = .Adodc1.Recordset(1)
    'A = .Adodc1.Recordset(6)
    'Ix = .Adodc1.Recordset(8)
    'Iy = .Adodc1.Recordset(11)


    'Densitat = .Adodc2.Recordset(2)
    'E = .Adodc2.Recordset(3) * 10000 * GGrav
    'Material = .Adodc2.Recordset(1)

    'h = .Adodc1.Recordset(2)
    'B = .Adodc1.Recordset(3)
    'DE = .Adodc1.Recordset(4)
    'DE1 = .Adodc1.Recordset(5)

    'TipusPerfil = 3
    'End With

End Sub

Sub AssignaDadesMaterial()
Dim E As Double, Nu As Double
Dim BarraEnCurs As Long
Dim Material As String
Dim Criteri As String
Dim Densitat As Double
Dim CoefDilatacioTermica As Double
'On Error Resume Next

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs
    With frmTipusBarra
    E = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).E
    Material = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Material

        If Material = "" Then
        Select Case EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil
        Case 1
        Material = EstructuraVis.Preferencies.MaterialPerDefecteSeccionsNoMetaliques
        Case 2
        Material = EstructuraVis.Preferencies.MaterialPerDefecteSeccionsNoMetaliques
        Case Is > 3
        Material = EstructuraVis.Preferencies.MaterialPerDefecteSeccionsMetaliques
        Case Else
```

```
    End Select
    End If


Criteri = "Nom LIKE " & "'" & Material & "'"

If Not .Adodc2.Recordset.EOF Then
.Adodc2.Recordset.MoveFirst
.Adodc2.Recordset.Find Criteri
End If

If .Adodc2.Recordset.EOF Then
.Adodc2.Recordset.MoveFirst
.Adodc2.Recordset.Find Criteri
    If .Adodc2.Recordset.EOF Then
    .Adodc2.Recordset.MoveFirst

    End If
End If




.DataCombo2.Text = Material



E = .Adodc2.Recordset(3)
Nu = .Adodc2.Recordset(4)
Material = .Adodc2.Recordset(1)
Densitat = .Adodc2.Recordset(2)
CoefDilatacioTermica = .Adodc2.Recordset(5)



.DataCombo2.Text = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Material
End With

With EstructuraVis.ColBarra.Item(BarraEnCurs)
.E = E
.Nu = Nu
.Material = Material
.Densitat = Densitat
.CoefDilatacioTermica = CoefDilatacioTermica
End With
```

```
End Sub


Sub AssignaDadesPerfilMetalic()

Dim BarraEnCurs As Long
Dim Serie As String
Dim Exemplar As String
Dim Criteri As String

'On Error Resume Next
BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs
   With frmTipusBarra
   Serie = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Serie
   Exemplar = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Denominacio

   If Serie = "" Then Serie = "Sèrie HEB"


      Select Case Serie

         Case "Sèrie IPN"
         .Adodc1.RecordSource = "select * from perfilsipn"

         Case "Sèrie IPE"
         .Adodc1.RecordSource = "select * from perfilsipe"

         Case "Sèrie HEB"
         .Adodc1.RecordSource = "select * from perfilsheb"

         Case "Sèrie HEA"
         .Adodc1.RecordSource = "select * from perfilshea"

         Case "Sèrie HEM"
         .Adodc1.RecordSource = "select * from perfilshem"

         Case "Sèrie UPN"
         .Adodc1.RecordSource = "select * from perfilsupn"

         Case "Sèrie L"
         .Adodc1.RecordSource = "select * from perfilsl"

         Case "Usuari"
         .Adodc1.RecordSource = "select * from perfilsdus"

      End Select
```

```
    .Combo1.Text = Serie
    .Combo1.Refresh

   .Adodc1.Refresh

   '.DataCombo1.DataField = "Nom"
   '.DataCombo1.Refresh
   '.DataCombo1.DataField = ""

   Criteri = "Nom LIKE " & "'" & Exemplar & "'"

   If Not .Adodc1.Recordset.EOF Then
   '.Adodc1.Recordset.MoveFirst
   .Adodc1.Recordset.Find Criteri
   End If

   If .Adodc1.Recordset.EOF Then
   .Adodc1.Recordset.MoveFirst
   .Adodc1.Recordset.Find Criteri
      If .Adodc1.Recordset.EOF Then
      .Adodc1.Recordset.MoveFirst

      End If
   End If

   With EstructuraVis.ColBarra.Item(Str(BarraEnCurs))
   .Denominacio = frmTipusBarra.Adodc1.Recordset(1)
   .A = frmTipusBarra.Adodc1.Recordset(6) / 10000
   .Ix = frmTipusBarra.Adodc1.Recordset(8) / 100000000


   .Iy = frmTipusBarra.Adodc1.Recordset(11) / 100000000
   .Ip = frmTipusBarra.Adodc1.Recordset(16) / 100000000

   .h = frmTipusBarra.Adodc1.Recordset(2) / 1000
   .B = frmTipusBarra.Adodc1.Recordset(3) / 1000
   .DE = frmTipusBarra.Adodc1.Recordset(4) / 1000
   .DE1 = frmTipusBarra.Adodc1.Recordset(5) / 1000

   End With


   End With

End Sub


Sub BorrarBarraEnCursFTP()
Dim BarraEnCurs As String
BarraEnCurs = Str(EstructuraVis.AuxFormularis.BarraEnCurs)
EstructuraVis.ColBarra.Remove (BarraEnCurs)
```

```
Principal.ActualitzaQuePotRenderitzarseIQueNo
Unload frmTipusBarra

End Sub


Sub ValidaDadesFrmSeccioGenerica()
Dim BarraEnCurs As Long

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs



With frmSeccioGenerica

EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Ix = Val(.Text1.Text) / 100000000
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Iy = Val(.Text2.Text) / 100000000
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Ip = Val(.Text3.Text) / 100000000
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).A = Val(.Text4.Text) / 10000
End With
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).TipusPerfil = 9
Unload frmSeccioGenerica
'Call DibuixaDadesPerfilGeneric
End Sub


Sub DibuixaDadesPerfilGeneric()
Dim BarraEnCurs As Long
Dim Ix As Double, Iy As Double, Ip As Double, A As Double, Wx As Double, Wy As Double

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs


Ix = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Ix * 100000000
Iy = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Iy * 100000000
Ip = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).Ip * 100000000
A = EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).A * 10000
With frmTipusBarra.Picture1
   .Cls
   .CurrentX = -50
   .CurrentY = 60
   frmTipusBarra.Picture1.Print " Dades perfil genèric:"
   .CurrentX = -50
   .CurrentY = 40
   frmTipusBarra.Picture1.Print "Ix: " & Ix & " cm^4"
   .CurrentX = -50
   .CurrentY = 20
   frmTipusBarra.Picture1.Print "Iy: " & Iy & " cm^4"
   .CurrentX = -50
   .CurrentY = 0
   frmTipusBarra.Picture1.Print "Ip: " & Ip & " cm^4"
```

```
    .CurrentX = -50
    .CurrentY = -20
    frmTipusBarra.Picture1.Print "A: " & A & " cm^2"



End With
End Sub

Sub CarregaDadesFormulariPerfilGeneric()
Dim BarraEnCurs As Long

BarraEnCurs = EstructuraVis.AuxFormularis.BarraEnCurs


With EstructuraVis.ColBarra.Item(Str(BarraEnCurs))
frmSeccioGenerica.Text1.Text = Format(.Ix * 100000000#, "0.00")
frmSeccioGenerica.Text2.Text = Format(.Iy * 100000000#, "0.00")
frmSeccioGenerica.Text3.Text = Format(.Ip * 100000000#, "0.00")
frmSeccioGenerica.Text4.Text = Format(.A * 10000, "0.00")
End With


End Sub
```

B. Representació gràfica

```
Option Explicit
'No me'l deixa definir en la classe CDirectXData

Public Const TL_FVF = (D3DFVF_XYZ Or D3DFVF_DIFFUSE Or D3DFVF_SPECULAR Or
D3DFVF_TEX1)

Public Type TLVERTEX
    X As Single
    Y As Single
    Z As Single
    Color As Long
    Specular As Long
    Tu As Single
    Tv As Single
End Type

Private m_VertexsDePissarra() As TLVERTEX
Private m_VertexsDePissarra2() As TLVERTEX
Private m_VertexsDePissarra3() As TLVERTEX
Private m_VertexsDePissarra4() As TLVERTEX
Private m_VertexsDePissarra5() As TLVERTEX

Private m_VertexsDePissarraA() As TLVERTEX
Private m_VertexsDePissarraA2() As TLVERTEX
```

```
Private m_VertexsDePissarraA3() As TLVERTEX
Private m_VertexsDePissarraA4() As TLVERTEX
Private m_VertexsDePissarraA5() As TLVERTEX


Private m_VertexsDePissarraS() As TLVERTEX
Private m_VertexsDePissarraS2() As TLVERTEX
Private m_VertexsDePissarraS3() As TLVERTEX
Private m_VertexsDePissarraS4() As TLVERTEX
Private m_VertexsDePissarraS5() As TLVERTEX



Private m_VertexsDeBarres() As TLVERTEX

Private m_VertexsDeNodes() As TLVERTEX
Private m_VertexsDeLiniesAuxiliarsDeNodes() As TLVERTEX

Private m_VertexsDeRestriccions() As TLVERTEX

Private m_VertexsDeReixeta() As TLVERTEX
Private m_VertexsDeReixetaMesProper(1 To 14) As TLVERTEX

Private m_VertexsDeEixos(1 To 9) As TLVERTEX
Private m_VertexsDeCapsDeEixos(1 To 6) As TLVERTEX

Private m_VertexsDeEixosLocals() As TLVERTEX
Private m_VertexsDeCapsDeEixosLocals() As TLVERTEX

Private m_VertexsDeForces() As TLVERTEX
Private m_VertexsDeCapsDeForces() As TLVERTEX

Private m_VertexsDeForcesBarra() As TLVERTEX
Private m_VertexsDeCapsDeForcesBarra() As TLVERTEX

Private m_VertexsDeLiniaAuxiliarIntroduccioBarres(1 To 2) As TLVERTEX

Private m_VertexsDeCarregaRepartida() As TLVERTEX
Private m_VertexsDePuntersDeCarregaRepartida() As TLVERTEX

Private m_VertexsDePerfilsDeConsulta() As TLVERTEX

Private m_VertexsDeArticulacions() As TLVERTEX

Private m_VertexsDeMomentsNode() As TLVERTEX
Private m_VertexsDeCapsDeMomentsNode() As TLVERTEX

Sub ActualitzaTotsElsBuffers()
'DMdlDirectX.RessetejaTotsElsBuffers
DMdlDirectX.AssignaGeometriaEixosMillorat
DMdlDirectX.AssignaGeometriaBarresaTotesLesVistes
```

```
DMdlDirectX.AssignaGeometriaForcesPuntualsATotesLesVistes
DMdlDirectX.AssignaGeometriaNodesATotesLesVistes
DMdlDirectX.AssignaGeometriaRestriccionsaTotesLesVistes
DMdlDirectX.AssignaGeometriaCarreguesRepartidesATotesLesVistes
DMdlDirectX.AssignaGeometriaEixosLocals
DMdlDirectX.AssignaGeometriaConsultaDePerfil
DMdlDirectX.AssignaGeometriaArticulacionsATotsElsBuffers
DMdlDirectX.AssignaGeometriaForcesBarraATotesLesVistes
DMdlDirectX.AssignaGeometriaMomentsNodalsATotesLesVistes
End Sub

Sub AssignaGeometriaEixosMillorat()
Dim CD As Double 'Corrector de Distància
Dim TamanyEixos As Double


Set EstructuraVis.DadesDirectX.VBufferEixosV3 = Nothing
TamanyEixos = EstructuraVis.Preferencies.TamanyEixos

CD = DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs *
TamanyEixos
With EstructuraVis.Preferencies
m_VertexsDeEixos(1) = CreateLitVertex(0, 0, 0, .ColorEixX, .ColorEixX, 0, 0)
m_VertexsDeEixos(2) = CreateLitVertex(1 * CD, 0, 0, .ColorEixXCap, .ColorEixXCap, 0, 0)
m_VertexsDeEixos(3) = CreateLitVertex(0, 0, 0.3 * CD, .ColorEixX, .ColorEixX, 0, 0)

m_VertexsDeEixos(4) = CreateLitVertex(0, 0, 0, .ColorEixY, .ColorEixY, 0, 0)
m_VertexsDeEixos(5) = CreateLitVertex(0, 1 * CD, 0, .ColorEixYCap, .ColorEixYCap, 0, 0)
m_VertexsDeEixos(6) = CreateLitVertex(0.3 * CD, 0, 0, .ColorEixY, .ColorEixY, 0, 0)

m_VertexsDeEixos(7) = CreateLitVertex(0, 0, 0, .ColorEixZ, .ColorEixZ, 0, 0)
m_VertexsDeEixos(8) = CreateLitVertex(0, 0, 1 * CD, .ColorEixZCap, .ColorEixZCap, 0, 0)
m_VertexsDeEixos(9) = CreateLitVertex(0, 0.3 * CD, 0, .ColorEixZ, .ColorEixZ, 0, 0)



m_VertexsDeCapsDeEixos(1) = CreateLitVertex(1 * CD, 0, 0, .ColorEixXCap,
.ColorEixXCap, 0, 0)
m_VertexsDeCapsDeEixos(2) = CreateLitVertex(0, 0, 0.3 * CD, .ColorEixX, .ColorEixX, 0,
0)

m_VertexsDeCapsDeEixos(3) = CreateLitVertex(0, 1 * CD, 0, .ColorEixYCap,
.ColorEixYCap, 0, 0)
m_VertexsDeCapsDeEixos(4) = CreateLitVertex(0.3 * CD, 0, 0, .ColorEixY, .ColorEixY, 0,
0)

m_VertexsDeCapsDeEixos(5) = CreateLitVertex(0, 0, 1 * CD, .ColorEixZCap,
.ColorEixZCap, 0, 0)
m_VertexsDeCapsDeEixos(6) = CreateLitVertex(0, 0.3 * CD, 0, .ColorEixZ, .ColorEixZ, 0, 0)

End With
```

```
        Set EstructuraVis.DadesDirectX.VBufferEixosV3 =
EstructuraVis.DadesDirectX.D3DDeviceV3.CreateVertexBuffer _
        (Len(m_VertexsDeEixos(1)) * 9, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferEixosV3, 0, _
        Len(m_VertexsDeEixos(1)) * 9, 0, m_VertexsDeEixos(1)


         Set EstructuraVis.DadesDirectX.VBufferDeCapsDeEixosV3 =
EstructuraVis.DadesDirectX.D3DDeviceV3.CreateVertexBuffer _
        (Len(m_VertexsDeCapsDeEixos(1)) * 6, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferDeCapsDeEixosV3,
0, _
        Len(m_VertexsDeCapsDeEixos(1)) * 6, 0, m_VertexsDeCapsDeEixos(1)


EstructuraVis.AuxFormularis.BufferEixosEmplenat = True

End Sub

Sub AssignaPuntDeVistaIAngleAPissarraRespecteVistaEnCurs()


   With EstructuraVis.Vis3
   EstructuraVis.DadesDirectX.AssignaPuntDeVistaDirectXAPissarra .AngleAlfa,
.AngleBeta, .DistanciaDelPuntDeVisioAlCentreDeGravetat, _
   EstructuraVis.Preferencies.CentreRotacioX, EstructuraVis.Preferencies.CentreRotacioY,
_
   EstructuraVis.Preferencies.CentreRotacioZ


   EstructuraVis.DadesDirectX.AssignaProjeccioDirectXAPissarra _
   EstructuraVis.Preferencies.AngleDeProjeccio, _
   EstructuraVis.Preferencies.DistanciaMinimaDeProjeccio, _
   EstructuraVis.Preferencies.DistanciaMaximaDeProjeccio, _
   FinestraCorresponentAlNumeroDeVistaAManipular.Height / _
   FinestraCorresponentAlNumeroDeVistaAManipular.Width
   End With
End Sub

Sub AturaRenderLopejant()
FrmMdi.Timer1.Enabled = False
End Sub

Sub DibuixaInformacioDelNodeDXALaVistaActiva()

Dim NumeroDeNodeAPresentar As Long
Dim XPresentar As Double, YPresentar As Double, ZPresentar As Double
Dim TextRect As RECT
Dim Color As Long
```

```
If EstructuraVis.ColNode.Count = 0 Then
Exit Sub
End If

Select Case Principal.FinestraCorresponentAlNumeroDeVistaAManipular.NumeroDeVista
Case 1
TextRect.Left = Vista1.XActual + 15
TextRect.Top = Vista1.YActual + 15
Case 2
TextRect.Left = Vista2.XActual + 15
TextRect.Top = Vista2.YActual + 15
Case 3
TextRect.Left = Vista3.XActual + 15
TextRect.Top = Vista3.YActual + 15
Case 4
TextRect.Left = Vista3.XActual + 15
TextRect.Top = Vista3.YActual + 15
End Select

TextRect.bottom = TextRect.Top + 40 'ARREGLA AIXO
TextRect.Right = TextRect.Left + 150



With EstructuraVis.ColNode
   If EstructuraVis.AuxFormularis.NodeMesProperPerConsultar = 0 Then
   EstructuraVis.AuxFormularis.NodeMesProperPerConsultar = 1
   End If

XPresentar = .Item(Str(EstructuraVis.AuxFormularis.NodeMesProperPerConsultar)).X
YPresentar = .Item(Str(EstructuraVis.AuxFormularis.NodeMesProperPerConsultar)).Y
ZPresentar = .Item(Str(EstructuraVis.AuxFormularis.NodeMesProperPerConsultar)).Z
End With

Color = D3DColorRGBA(150, 0, 0, 255)


Select Case Principal.FinestraCorresponentAlNumeroDeVistaAManipular.NumeroDeVista
Case 1
   EstructuraVis.DadesDirectX.D3DX.DrawText EstructuraVis.DadesDirectX.MainFontV1,
EstructuraVis.Preferencies.ColorTextAnglesAlfaIBeta, _
   XPresentar & ", " & YPresentar & ", " & ZPresentar, _
   TextRect, DT_TOP Or DT_LEFT
Case 2
   EstructuraVis.DadesDirectX.D3DX.DrawText EstructuraVis.DadesDirectX.MainFontV2,
EstructuraVis.Preferencies.ColorTextAnglesAlfaIBeta, _
   XPresentar & ", " & YPresentar & ", " & ZPresentar, _
   TextRect, DT_TOP Or DT_LEFT
Case 3 'FALTA CANVIAR EL COLOR ALS ALTRES
```

```
    EstructuraVis.DadesDirectX.D3DX.DrawText EstructuraVis.DadesDirectX.MainFontV3,
Color, _
    "Node " & EstructuraVis.AuxFormularis.NodeMesProperPerConsultar & vbCrLf & "(" &
XPresentar & ", " & YPresentar & ", " & ZPresentar & ")", _
   TextRect, DT_LEFT


Case 4
    EstructuraVis.DadesDirectX.D3DX.DrawText EstructuraVis.DadesDirectX.MainFontV4,
EstructuraVis.Preferencies.ColorTextAnglesAlfaIBeta, _
    XPresentar & ", " & YPresentar & ", " & ZPresentar, _
    TextRect, DT_TOP Or DT_LEFT
End Select


End Sub


Sub DibuixaInformacioDelNodeAEsborrarDXALaVistaActiva()

Dim NumeroDeNodeAPresentar As Long
Dim XPresentar As Double, YPresentar As Double, ZPresentar As Double
Dim TextRect As RECT
Dim Color As Long

Select Case Principal.FinestraCorresponentAlNumeroDeVistaAManipular.NumeroDeVista
Case 1
TextRect.Left = Vista1.XActual + 15
TextRect.Top = Vista1.YActual + 15
Case 2
TextRect.Left = Vista2.XActual + 15
TextRect.Top = Vista2.YActual + 15
Case 3
TextRect.Left = Vista3.XActual + 15
TextRect.Top = Vista3.YActual + 15
Case 4
TextRect.Left = Vista3.XActual + 15
TextRect.Top = Vista3.YActual + 15
End Select

TextRect.bottom = TextRect.Top + 40 'ARREGLA AIXO
TextRect.Right = TextRect.Left + 150

With EstructuraVis.ColNode
'XPresentar = .Item(Str(EstructuraVis.AuxFormularis.NodeMesProperPerConsultar)).X
'YPresentar = .Item(Str(EstructuraVis.AuxFormularis.NodeMesProperPerConsultar)).Y
'ZPresentar = .Item(Str(EstructuraVis.AuxFormularis.NodeMesProperPerConsultar)).Z
End With

Color = D3DColorRGBA(255, 0, 0, 255)
```

```
Select Case Principal.FinestraCorresponentAlNumeroDeVistaAManipular.NumeroDeVista
Case 1
   EstructuraVis.DadesDirectX.D3DX.DrawText EstructuraVis.DadesDirectX.MainFontV1,
EstructuraVis.Preferencies.ColorTextAnglesAlfaIBeta, _
   XPresentar & ", " & YPresentar & ", " & ZPresentar, _
   TextRect, DT_TOP Or DT_LEFT
Case 2
   EstructuraVis.DadesDirectX.D3DX.DrawText EstructuraVis.DadesDirectX.MainFontV2,
EstructuraVis.Preferencies.ColorTextAnglesAlfaIBeta, _
   XPresentar & ", " & YPresentar & ", " & ZPresentar, _
   TextRect, DT_TOP Or DT_LEFT
Case 3 'FALTA CANVIAR EL COLOR ALS ALTRES
   If EstructuraVis.AuxFormularis.NodeMesProperPerEsborrar = -1 Then
   EstructuraVis.DadesDirectX.D3DX.DrawText EstructuraVis.DadesDirectX.MainFontV3,
Color, _
   "No hi ha cap node", _
  TextRect, DT_LEFT
   Else
   EstructuraVis.DadesDirectX.D3DX.DrawText EstructuraVis.DadesDirectX.MainFontV3,
Color, _
   "Esborrar " & vbCrLf & "Node " &
EstructuraVis.AuxFormularis.NodeMesProperPerEsborrar & "?", _
  TextRect, DT_LEFT
   End If

Case 4
   EstructuraVis.DadesDirectX.D3DX.DrawText EstructuraVis.DadesDirectX.MainFontV4,
EstructuraVis.Preferencies.ColorTextAnglesAlfaIBeta, _
   XPresentar & ", " & YPresentar & ", " & ZPresentar, _
   TextRect, DT_TOP Or DT_LEFT
End Select


End Sub
Sub DibuixaNumeracioBarres()
Dim BarraEnCurs As Long, MaxBarra As Long
Dim Cent As D3DVECTOR
Dim VD As D3DVECTOR
Dim VN As D3DVECTOR
Dim VC As D3DVECTOR
Dim VProj As D3DVECTOR
Dim Text As String
MaxBarra = EstructuraVis.ColBarra.Count

   For BarraEnCurs = 1 To MaxBarra

      With EstructuraVis.ColBarra.Item(BarraEnCurs)
         VD = CreaVector(.X2 - .X1, .Y2 - .Y1, .Z2 - .Z1)
         D3DXVec3Normalize VD, VD
         VC = CreaVector(.VxOrto, .VyOrto, .VzOrto)
```

```
            D3DXVec3Normalize VC, VC
            D3DXVec3Cross VN, VD, VC
            D3DXVec3Normalize VN, VN

        VC.X = .X1 + VD.X * .L / 2
        VC.Y = .Y1 + VD.Y * .L / 2
        VC.Z = .Z1 + VD.Z * .L / 2

        Text = Str(.Numero)

        End With

        'PRIMER PROVO DE POSAR-HO AL CENTRE CENTRE
        VProj = EstructuraVis.DadesDirectX.ProjectaVectorEnFinestraEnCurs(VC)
        DMdlDirectX.EscriuTextEnPantallaIndicantCoordenadesFinestra3DX "Comic Sans", 8,
_
        (VProj.X), (VProj.Y), 20, 20, D3DColorRGBA(0, 0, 255, 255), Text, DT_TOP Or
DT_LEFT


    Next BarraEnCurs

End Sub



Sub DibuixaTextMaterialBarres()
Dim BarraEnCurs As Long, MaxBarra As Long
Dim Cent As D3DVECTOR
Dim VD As D3DVECTOR
Dim VN As D3DVECTOR
Dim VC As D3DVECTOR
Dim VProj As D3DVECTOR
Dim Text As String
MaxBarra = EstructuraVis.ColBarra.Count

    For BarraEnCurs = 1 To MaxBarra

        With EstructuraVis.ColBarra.Item(BarraEnCurs)
            VD = CreaVector(.X2 - .X1, .Y2 - .Y1, .Z2 - .Z1)
            D3DXVec3Normalize VD, VD
            VC = CreaVector(.VxOrto, .VyOrto, .VzOrto)
            D3DXVec3Normalize VC, VC
            D3DXVec3Cross VN, VD, VC
            D3DXVec3Normalize VN, VN

        VC.X = .X1 + VD.X * .L / 2
        VC.Y = .Y1 + VD.Y * .L / 2
        VC.Z = .Z1 + VD.Z * .L / 2

        Text = (.Material)
```

```
        End With

        'PRIMER PROVO DE POSAR-HO AL CENTRE CENTRE
        VProj = EstructuraVis.DadesDirectX.ProjectaVectorEnFinestraEnCurs(VC)
        VProj.Y = VProj.Y - 12: VProj.X = VProj.X + 10
        DMdlDirectX.EscriuTextEnPantallaIndicantCoordenadesFinestra3DX "Arial", 7, _
        (VProj.X), (VProj.Y), 50, 800, D3DColorRGBA(100, 0, 255, 255), Text, DT_TOP Or
DT_LEFT


    Next BarraEnCurs

End Sub

Sub DibuixaTextEixos()

Dim BarraEnCurs As Long

Dim VE As D3DVECTOR
Dim VProj As D3DVECTOR
Dim Text As String
Dim CD As Double


CD = DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs *
EstructuraVis.Preferencies.TamanyEixos

        'PRIMER
        VE.X = CD * 1.3
        VE.Y = 0
        VE.Z = 0
        Text = "x"
        VProj = EstructuraVis.DadesDirectX.ProjectaVectorEnFinestraEnCurs(VE)
        DMdlDirectX.EscriuTextEnPantallaIndicantCoordenadesFinestra3DX "Arial", 7, _
        (VProj.X), (VProj.Y), 20, 20, D3DColorRGBA(150, 0, 255, 255), Text, DT_TOP Or
DT_LEFT

        'SEGON
        VE.X = 0
        VE.Y = CD * 1.3
        VE.Z = 0
        Text = "y"
        VProj = EstructuraVis.DadesDirectX.ProjectaVectorEnFinestraEnCurs(VE)
        DMdlDirectX.EscriuTextEnPantallaIndicantCoordenadesFinestra3DX "Arial", 7, _
        (VProj.X), (VProj.Y), 20, 20, D3DColorRGBA(150, 0, 255, 255), Text, DT_TOP Or
DT_LEFT


        'TERCER
        VE.X = 0
```

```
      VE.Y = 0
      VE.Z = CD * 1.3
      Text = "z"
      VProj = EstructuraVis.DadesDirectX.ProjectaVectorEnFinestraEnCurs(VE)
      DMdlDirectX.EscriuTextEnPantallaIndicantCoordenadesFinestra3DX "Arial", 7, _
      (VProj.X), (VProj.Y), 20, 20, D3DColorRGBA(150, 0, 255, 255), Text, DT_TOP Or
DT_LEFT



End Sub



Sub DibuixaTextReferenciaBarres()
Dim BarraEnCurs As Long, MaxBarra As Long
Dim Cent As D3DVECTOR
Dim VD As D3DVECTOR
Dim VN As D3DVECTOR
Dim VC As D3DVECTOR
Dim VProj As D3DVECTOR
Dim Text As String
MaxBarra = EstructuraVis.ColBarra.Count

   For BarraEnCurs = 1 To MaxBarra

      With EstructuraVis.ColBarra.Item(BarraEnCurs)
         VD = CreaVector(.X2 - .X1, .Y2 - .Y1, .Z2 - .Z1)
         D3DXVec3Normalize VD, VD
         VC = CreaVector(.VxOrto, .VyOrto, .VzOrto)
         D3DXVec3Normalize VC, VC
         D3DXVec3Cross VN, VD, VC
         D3DXVec3Normalize VN, VN

      VC.X = .X1 + VD.X * .L / 2
      VC.Y = .Y1 + VD.Y * .L / 2
      VC.Z = .Z1 + VD.Z * .L / 2

      Text = (.Denominacio)

      End With

      'PRIMER PROVO DE POSAR-HO AL CENTRE CENTRE
      VProj = EstructuraVis.DadesDirectX.ProjectaVectorEnFinestraEnCurs(VC)
      VProj.Y = VProj.Y + 12: VProj.X = VProj.X + 10
      DMdlDirectX.EscriuTextEnPantallaIndicantCoordenadesFinestra3DX "Arial", 7, _
      (VProj.X), (VProj.Y), 50, 800, D3DColorRGBA(0, 175, 0, 255), Text, DT_TOP Or
DT_LEFT


   Next BarraEnCurs
```

```
End Sub



Sub DibuixaNumeracioNodes()
Dim NodeEnCurs As Long, MaxNode As Long
Dim VC As D3DVECTOR
Dim VProj As D3DVECTOR
Dim Text As String
MaxNode = EstructuraVis.ColNode.Count

   For NodeEnCurs = 1 To MaxNode

      With EstructuraVis.ColNode.Item(NodeEnCurs)

      VC.X = .X
      VC.Y = .Y
      VC.Z = .Z

      Text = Str(.Numero)

      End With

      'PRIMER PROVO DE POSAR-HO AL CENTRE CENTRE
      VProj = EstructuraVis.DadesDirectX.ProjectaVectorEnFinestraEnCurs(VC)
      VProj.X = VProj.X + 10
      DMdlDirectX.EscriuTextEnPantallaIndicantCoordenadesFinestra3DX "Arial", 8, _
      (VProj.X), (VProj.Y), 20, 20, D3DColorRGBA(0, 0, 0, 255), Text, DT_TOP Or
DT_LEFT

   Next NodeEnCurs

End Sub



Sub DibuixaValorsForcesNodals()
Dim NodeEnCurs As Long, MaxNode As Long
Dim VC As D3DVECTOR
Dim VProj As D3DVECTOR
Dim Text As String
Dim AccioActivada As Long
Dim Modul As Double
Dim ValorMaximDeForça As Double
Dim SumaDeTotsElsComponents As Double
```

```
MaxNode = EstructuraVis.ColNode.Count

AccioActivada = EstructuraVis.AuxFormularis.AccioActivada


      'DETERMINO LA FORÇA MÀXIMA FENT UN RECORREGUT PER TOTS ELS
NODES

   For NodeEnCurs = 1 To MaxNode

      If EstructuraVis.ColNode.Item(NodeEnCurs).ExisteixLaAccioNodal(Str(AccioActivada))
= True Then


            With
EstructuraVis.ColNode.Item(NodeEnCurs).ColSolicitacioNodal.Item(Str(AccioActivada))

            SumaDeTotsElsComponents = (.Fx ^ 2 + .Fy ^ 2 + .Fz ^ 2) ^ 0.5

            If Abs(ValorMaximDeForça) < SumaDeTotsElsComponents Then
            ValorMaximDeForça = (.Fx ^ 2 + .Fy ^ 2 + .Fz ^ 2) ^ 0.5
            End If
            End With


   End If

   Next NodeEnCurs


   If ValorMaximDeForça = 0 Then ValorMaximDeForça = 1

   'FI DETERMINACIÓ DE LA FORÇA MÀXIMA

   For NodeEnCurs = 1 To MaxNode

      With EstructuraVis.ColNode.Item(NodeEnCurs)
      If .ExisteixLaAccioNodal(Str(AccioActivada)) = False Then GoTo Etiqueta

      VC.X = .ColSolicitacioNodal.Item(Str(AccioActivada)).Fx * _
      DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs * _
      EstructuraVis.Preferencies.DimensioDeLaLongitudDeLaLiniaDeForces / _
      ValorMaximDeForça

      VC.Y = .ColSolicitacioNodal.Item(Str(AccioActivada)).Fy _
      * DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs _
      * EstructuraVis.Preferencies.DimensioDeLaLongitudDeLaLiniaDeForces / _
      ValorMaximDeForça

      VC.Z = .ColSolicitacioNodal.Item(Str(AccioActivada)).Fz _
```

```
        * DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs _
        * EstructuraVis.Preferencies.DimensioDeLaLongitudDeLaLiniaDeForces / _
        ValorMaximDeForça

        Modul = D3DXVec3Length(VC) / 1000


        Modul =
D3DXVec3Length(CreaVector(.ColSolicitacioNodal.Item(Str(AccioActivada)).Fx, _
        .ColSolicitacioNodal.Item(Str(AccioActivada)).Fy, _
        .ColSolicitacioNodal.Item(Str(AccioActivada)).Fz)) / 1000

        If Modul < 0.00001 Then GoTo Etiqueta '1 MILIGRAM!'

        Text = Format(Modul, "##.00") & " kN"

        VC.X = -VC.X + .X
        VC.Y = -VC.Y + .Y
        VC.Z = -VC.Z + .Z

        End With

        'PRIMER PROVO DE POSAR-HO AL CENTRE CENTRE
        VProj = EstructuraVis.DadesDirectX.ProjectaVectorEnFinestraEnCurs(VC)
        VProj.X = VProj.X + 10
        DMdlDirectX.EscriuTextEnPantallaIndicantCoordenadesFinestra3DX "Arial", 7, _
        (VProj.X), (VProj.Y), 50, 20, D3DColorRGBA(0, 0, 0, 255), Text, DT_TOP Or
DT_LEFT
Etiqueta:
    Next NodeEnCurs

End Sub


Sub DibuixaValorsMomentsNodals()
Dim NodeEnCurs As Long, MaxNode As Long
Dim VC As D3DVECTOR
Dim VProj As D3DVECTOR
Dim Text As String
Dim AccioActivada As Long
Dim Modul As Double
Dim ValorMaximDeForça As Double
Dim SumaDeTotsElsComponents As Double


MaxNode = EstructuraVis.ColNode.Count

AccioActivada = EstructuraVis.AuxFormularis.AccioActivada
```

```
        'DETERMINO LA FORÇA MÀXIMA FENT UN RECORREGUT PER TOTS ELS
NODES

   For NodeEnCurs = 1 To MaxNode

      If EstructuraVis.ColNode.Item(NodeEnCurs).ExisteixLaAccioNodal(Str(AccioActivada))
= True Then


            With
EstructuraVis.ColNode.Item(NodeEnCurs).ColSolicitacioNodal.Item(Str(AccioActivada))

            SumaDeTotsElsComponents = (.Mx ^ 2 + .My ^ 2 + .Mz ^ 2) ^ 0.5

            If Abs(ValorMaximDeForça) < SumaDeTotsElsComponents Then
            ValorMaximDeForça = (.Mx ^ 2 + .My ^ 2 + .Mz ^ 2) ^ 0.5
            End If
            End With


   End If

   Next NodeEnCurs


   If ValorMaximDeForça = 0 Then ValorMaximDeForça = 1

   'FI DETERMINACIÓ DE LA FORÇA MÀXIMA

   For NodeEnCurs = 1 To MaxNode

      With EstructuraVis.ColNode.Item(NodeEnCurs)
      If .ExisteixLaAccioNodal(Str(AccioActivada)) = False Then GoTo Etiqueta

      VC.X = .ColSolicitacioNodal.Item(Str(AccioActivada)).Mx * _
      DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs * _
      EstructuraVis.Preferencies.DimensioDeLaLongitudDeLaLiniaDeForces / _
      ValorMaximDeForça

      VC.Y = .ColSolicitacioNodal.Item(Str(AccioActivada)).My _
      * DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs _
      * EstructuraVis.Preferencies.DimensioDeLaLongitudDeLaLiniaDeForces / _
      ValorMaximDeForça

      VC.Z = .ColSolicitacioNodal.Item(Str(AccioActivada)).Mz _
      * DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs _
      * EstructuraVis.Preferencies.DimensioDeLaLongitudDeLaLiniaDeForces / _
      ValorMaximDeForça

      Modul = D3DXVec3Length(VC) / 1000
```

```
        Modul =
D3DXVec3Length(CreaVector(.ColSolicitacioNodal.Item(Str(AccioActivada)).Mx, _
        .ColSolicitacioNodal.Item(Str(AccioActivada)).My, _
        .ColSolicitacioNodal.Item(Str(AccioActivada)).Mz)) / 1000

        Text = Format(Modul, "##.00") & " kN·m"

        VC.X = -VC.X + .X
        VC.Y = -VC.Y + .Y
        VC.Z = -VC.Z + .Z

        End With

        'PRIMER PROVO DE POSAR-HO AL CENTRE CENTRE
        VProj = EstructuraVis.DadesDirectX.ProjectaVectorEnFinestraEnCurs(VC)
        VProj.X = VProj.X + 10
        DMdlDirectX.EscriuTextEnPantallaIndicantCoordenadesFinestra3DX "Arial", 7, _
        (VProj.X), (VProj.Y), 50, 20, D3DColorRGBA(0, 0, 0, 255), Text, DT_TOP Or
DT_LEFT
Etiqueta:
    Next NodeEnCurs

End Sub


Function DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs() As Double
Dim V1 As D3DVECTOR
Dim V2 As D3DVECTOR
Dim V3 As D3DVECTOR
Dim Modul As Double
Dim Alfa As Double, Beta As Double, Dist As Double


'Alfa = Principal.VistaCorresponentAlNumeroDeVistaAManipular.AngleAlfa
'Beta = Principal.VistaCorresponentAlNumeroDeVistaAManipular.AngleBeta
'Dist =
Principal.VistaCorresponentAlNumeroDeVistaAManipular.DistanciaDelPuntDeVisioAlCentre
DeGravetat

Alfa = EstructuraVis.Vis3.AngleAlfa
Beta = EstructuraVis.Vis3.AngleBeta
Dist = EstructuraVis.Vis3.DistanciaDelPuntDeVisioAlCentreDeGravetat



V1 = CreaVector(Dist * Cos(Alfa) * Cos(Beta), _
Dist * Sin(Alfa), _
Dist * Cos(Alfa) * Sin(Beta))
'V2 = CreaVector(EstructuraVis.Preferencies.CentreRotacioX, _
'EstructuraVis.Preferencies.CentreRotacioY, _
```

```
'EstructuraVis.Preferencies.CentreRotacioZ)
'D3DXVec3Add V3, V1, V2
Modul = D3DXVec3Length(V1)

DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs = Modul
End Function

Sub RenderPissarra()

    'RENDERITZEM El frmPissarra
    If EstructuraVis.ColBarra.Count > 0 Then
    EstructuraVis.DadesDirectX.D3DDevicePissarra.Clear 0, ByVal 0, D3DCLEAR_TARGET
Or D3DCLEAR_ZBUFFER, 0, 1#, 0  '//Clear the screen white
    EstructuraVis.DadesDirectX.D3DDevicePissarra.BeginScene

    EstructuraVis.DadesDirectX.D3DDevicePissarra.SetVertexShader TL_FVF

    'BIGUES SECCIÓ RECTANGULAR DE FORMIGÓ
    EstructuraVis.DadesDirectX.D3DDevicePissarra.SetTexture 0,
EstructuraVis.DadesDirectX.TexturaFormigo
    EstructuraVis.DadesDirectX.D3DDevicePissarra.SetStreamSource 0,
EstructuraVis.DadesDirectX.vbufferpissarra, Len(m_VertexsDePissarra(1))
    EstructuraVis.DadesDirectX.D3DDevicePissarra.DrawPrimitive D3DPT_TRIANGLELIST,
0, (EstructuraVis.ColBarra.Count * 12)
    'BIGUES SECCIÓ H,I DE FORMIGÓ
    EstructuraVis.DadesDirectX.D3DDevicePissarra.SetStreamSource 0,
EstructuraVis.DadesDirectX.VBufferPissarra2, Len(m_VertexsDePissarra2(1))
    EstructuraVis.DadesDirectX.D3DDevicePissarra.DrawPrimitive D3DPT_TRIANGLELIST,
0, (EstructuraVis.ColBarra.Count * 6)
    'PILARS CIRCULARS DE FORMIGÓ
    EstructuraVis.DadesDirectX.D3DDevicePissarra.SetStreamSource 0,
EstructuraVis.DadesDirectX.VBufferPissarra3, Len(m_VertexsDePissarra3(1))
    EstructuraVis.DadesDirectX.D3DDevicePissarra.DrawPrimitive D3DPT_TRIANGLELIST,
0, (EstructuraVis.ColBarra.Count * 80)
    'BIGUES U FORMIGÓ
    EstructuraVis.DadesDirectX.D3DDevicePissarra.SetStreamSource 0,
EstructuraVis.DadesDirectX.VBufferPissarra4, Len(m_VertexsDePissarra4(1))
    EstructuraVis.DadesDirectX.D3DDevicePissarra.DrawPrimitive D3DPT_TRIANGLELIST,
0, (EstructuraVis.ColBarra.Count * 6)
    'BIGUES L FORMIGÓ
    EstructuraVis.DadesDirectX.D3DDevicePissarra.SetStreamSource 0,
EstructuraVis.DadesDirectX.VBufferPissarra5, Len(m_VertexsDePissarra5(1))
    EstructuraVis.DadesDirectX.D3DDevicePissarra.DrawPrimitive D3DPT_TRIANGLELIST,
0, (EstructuraVis.ColBarra.Count * 4)



    'BIGUES SECCIÓ RECTANGULAR ACER
    EstructuraVis.DadesDirectX.D3DDevicePissarra.SetTexture 0,
EstructuraVis.DadesDirectX.TexturaAcer
```

```
    EstructuraVis.DadesDirectX.D3DDevicePissarra.SetStreamSource 0,
EstructuraVis.DadesDirectX.VBufferPissarraS, Len(m_VertexsDePissarraS(1))
    EstructuraVis.DadesDirectX.D3DDevicePissarra.DrawPrimitive D3DPT_TRIANGLELIST,
0, (EstructuraVis.ColBarra.Count * 12)
    'BIGUES SECCIÓ H,I ACER
    EstructuraVis.DadesDirectX.D3DDevicePissarra.SetStreamSource 0,
EstructuraVis.DadesDirectX.VBufferPissarraS2, Len(m_VertexsDePissarraS2(1))
    EstructuraVis.DadesDirectX.D3DDevicePissarra.DrawPrimitive D3DPT_TRIANGLELIST,
0, (EstructuraVis.ColBarra.Count * 6)
    'BIGUES U,ACER
    EstructuraVis.DadesDirectX.D3DDevicePissarra.SetStreamSource 0,
EstructuraVis.DadesDirectX.VBufferPissarraS4, Len(m_VertexsDePissarraS4(1))
    EstructuraVis.DadesDirectX.D3DDevicePissarra.DrawPrimitive D3DPT_TRIANGLELIST,
0, (EstructuraVis.ColBarra.Count * 6)
    'BIGUES L,ACER
    EstructuraVis.DadesDirectX.D3DDevicePissarra.SetStreamSource 0,
EstructuraVis.DadesDirectX.VBufferPissarraS5, Len(m_VertexsDePissarraS5(1))
    EstructuraVis.DadesDirectX.D3DDevicePissarra.DrawPrimitive D3DPT_TRIANGLELIST,
0, (EstructuraVis.ColBarra.Count * 4)


    'BIGUES SECCIÓ RECTANGULAR ALTRE MATERIAL
    EstructuraVis.DadesDirectX.D3DDevicePissarra.SetTexture 0,
EstructuraVis.DadesDirectX.TexturaAltres
    EstructuraVis.DadesDirectX.D3DDevicePissarra.SetStreamSource 0,
EstructuraVis.DadesDirectX.VBufferPissarraA, Len(m_VertexsDePissarraA(1))
    EstructuraVis.DadesDirectX.D3DDevicePissarra.DrawPrimitive D3DPT_TRIANGLELIST,
0, (EstructuraVis.ColBarra.Count * 12)
    'BIGUES U ALTRE MATERIAL
    EstructuraVis.DadesDirectX.D3DDevicePissarra.SetStreamSource 0,
EstructuraVis.DadesDirectX.VBufferPissarraA4, Len(m_VertexsDePissarraA4(1))
    EstructuraVis.DadesDirectX.D3DDevicePissarra.DrawPrimitive D3DPT_TRIANGLELIST,
0, (EstructuraVis.ColBarra.Count * 6)
    'BIGUES L, ALTRE MATERIAL
    EstructuraVis.DadesDirectX.D3DDevicePissarra.SetStreamSource 0,
EstructuraVis.DadesDirectX.VBufferPissarraA5, Len(m_VertexsDePissarraA5(1))
    EstructuraVis.DadesDirectX.D3DDevicePissarra.DrawPrimitive D3DPT_TRIANGLELIST,
0, (EstructuraVis.ColBarra.Count * 4)




    EstructuraVis.DadesDirectX.D3DDevicePissarra.EndScene

    EstructuraVis.DadesDirectX.D3DDevicePissarra.Present ByVal 0, ByVal 0, 0, ByVal 0

    DoEvents

    End If
```

```
   'FI RENDERITZAT PISSARRA
End Sub



Function Signe(Valor As Double) As Long
If Valor < 0 Then
Signe = -1
ElseIf Valor = 0 Then
Signe = 0
ElseIf Valor > 0 Then
Signe = 1
End If
End Function

Sub XXXXXXX()

End Sub


Sub AssignaGeometriaReixeta()
Dim iReixeta As Long
Dim jReixeta As Long
Dim Max As Double
Dim Ample As Double
Dim Contador As Double
Dim X As Integer, Y As Integer, Z As Integer
Dim TipusDeReixeta As Integer

Max = EstructuraVis.Preferencies.DimensioReixeta
Ample = EstructuraVis.Preferencies.AmpleReixeta
Contador = 1


   If EstructuraVis.Preferencies.XReixetaDir = 1 Then
   X = 0
   Else: X = 1
   End If
   If EstructuraVis.Preferencies.YReixetaDir = 1 Then
   Y = 0
   Else: Y = 1
   End If
   If EstructuraVis.Preferencies.ZReixetaDir = 1 Then
   Z = 0
   Else: Z = 1
   End If


   If X = 1 And Y = 1 And Z = 0 Then
   TipusDeReixeta = 1
   End If
```

```
    If X = 1 And Y = 0 And Z = 1 Then
    TipusDeReixeta = 2
    End If
    If X = 0 And Y = 1 And Z = 1 Then
    TipusDeReixeta = 3
    End If

ReDim m_VertexsDeReixeta(1 To (Max * 2 + 1) ^ 2)

    For iReixeta = -Max To Max
       For jReixeta = -Max To Max

       Select Case TipusDeReixeta

          Case 1
          m_VertexsDeReixeta(Contador) = CreateLitVertex(Ample * iReixeta, Ample *
jReixeta, EstructuraVis.Preferencies.ZReixeta, EstructuraVis.Preferencies.ColorReixeta, 1,
0, 0)
          Case 2
          m_VertexsDeReixeta(Contador) = CreateLitVertex(Ample * iReixeta,
EstructuraVis.Preferencies.YReixeta, Ample * jReixeta,
EstructuraVis.Preferencies.ColorReixeta, 1, 0, 0)
          Case 3
          m_VertexsDeReixeta(Contador) =
CreateLitVertex(EstructuraVis.Preferencies.XReixeta, Ample * iReixeta, Ample * jReixeta,
EstructuraVis.Preferencies.ColorReixeta, 1, 0, 0)
          End Select

       Contador = Contador + 1
       Next jReixeta
    Next iReixeta
    'We ensure that Vbuffers as empty

     Set EstructuraVis.DadesDirectX.VBufferReixetaV1 = Nothing
     Set EstructuraVis.DadesDirectX.VBufferReixetaV2 = Nothing
     Set EstructuraVis.DadesDirectX.VBufferReixetaV3 = Nothing
     Set EstructuraVis.DadesDirectX.VBufferReixetaV4 = Nothing



    'I create the vertex buffers
     'Set EstructuraVis.DadesDirectX.VBufferReixetaV1 =
EstructuraVis.DadesDirectX.D3DDeviceV1.CreateVertexBuffer _
 (Len(m_VertexsDeReixeta(1)) * Contador, 0, TL_FVF, D3DPOOL_DEFAULT)
     'Set EstructuraVis.DadesDirectX.VBufferReixetaV2 =
EstructuraVis.DadesDirectX.D3DDeviceV2.CreateVertexBuffer _
 (Len(m_VertexsDeReixeta(1)) * Contador, 0, TL_FVF, D3DPOOL_DEFAULT)
     Set EstructuraVis.DadesDirectX.VBufferReixetaV3 =
EstructuraVis.DadesDirectX.D3DDeviceV3.CreateVertexBuffer _
 (Len(m_VertexsDeReixeta(1)) * Contador, 0, TL_FVF, D3DPOOL_DEFAULT)
```

```
    'Set EstructuraVis.DadesDirectX.VBufferReixetaV4 =
EstructuraVis.DadesDirectX.D3DDeviceV4.CreateVertexBuffer _
 (Len(m_VertexsDeReixeta(1)) * Contador, 0, TL_FVF, D3DPOOL_DEFAULT)

   'I fill the created vertex with the data

        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferReixetaV1, 0, _
        Len(m_VertexsDeReixeta(1)) * Contador, 0, m_VertexsDeReixeta(1)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferReixetaV2, 0, _
        Len(m_VertexsDeReixeta(1)) * Contador, 0, m_VertexsDeReixeta(1)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferReixetaV3, 0, _
        Len(m_VertexsDeReixeta(1)) * Contador, 0, m_VertexsDeReixeta(1)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferReixetaV4, 0, _
        Len(m_VertexsDeReixeta(1)) * Contador, 0, m_VertexsDeReixeta(1)


   EstructuraVis.AuxFormularis.BufferReixetaEmplenat = True
End Sub

Sub AssignaGeometriaLiniaAuxiliarIntroduccioBarres(ByVal XPantalla As Double, ByVal
YPantalla As Double)

Dim V1 As D3DVECTOR, V2 As D3DVECTOR, V3 As D3DVECTOR

Dim AntiProjeccioX As Single, AntiProjeccioY As Single
Dim Auxil As Double
Dim X1 As Double, Y1 As Double, Z1 As Double
Dim NodeInicial As Long

With EstructuraVis.AuxFormularis
NodeInicial = .NodeInicialLiniaAuxiliar
End With


If IntrodNodes.ElNodeExisteix(NodeInicial) = False Then Exit Sub

With EstructuraVis.AuxFormularis
X1 = EstructuraVis.ColNode.Item(Str(NodeInicial)).X
Y1 = EstructuraVis.ColNode.Item(Str(NodeInicial)).Y
Z1 = EstructuraVis.ColNode.Item(Str(NodeInicial)).Z



End With
'HEM DE DISCRIMINAR RESPECTE LA VISTA SOBRE LA QUAL ESTEM TREBALLANT

If EstructuraVis.ColNode.Count = 0 Then Exit Sub

'x1, y1, z1 són les coordenades del node
m_VertexsDeLiniaAuxiliarIntroduccioBarres(1) = CreateLitVertex((X1), (Y1), (Z1),
QBColor(12), 1, 0, 0)
```

```
V1 = CreaVector((XPantalla), (YPantalla), 0)


Select Case FinestraCorresponentAlNumeroDeVistaAManipular.NumeroDeVista
Case 1
EstructuraVis.DadesDirectX.AntiProjectaVectorDX V2, V1, 1
Case 2
EstructuraVis.DadesDirectX.AntiProjectaVectorDX V2, V1, 2
Case 3
EstructuraVis.DadesDirectX.AntiProjectaVectorDX V2, V1, 3
Case 4
EstructuraVis.DadesDirectX.AntiProjectaVectorDX V2, V1, 4
End Select

m_VertexsDeLiniaAuxiliarIntroduccioBarres(2) = CreateLitVertex(V2.X, V2.Y, V2.Z,
QBColor(12), 1, 0, 0)


'Set EstructuraVis.DadesDirectX.VBufferBarraAuxiliarV1 =
EstructuraVis.DadesDirectX.D3DDeviceV1.CreateVertexBuffer _
(Len(m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)) * 2, 0, TL_FVF, D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferBarraAuxiliarV1, 0, _
Len(m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)) * 2, 0,
m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)

'Set EstructuraVis.DadesDirectX.VBufferBarraAuxiliarV2 =
EstructuraVis.DadesDirectX.D3DDeviceV2.CreateVertexBuffer _
(Len(m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)) * 2, 0, TL_FVF, D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferBarraAuxiliarV2, 0, _
Len(m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)) * 2, 0,
m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)

Set EstructuraVis.DadesDirectX.VBufferBarraAuxiliarV3 =
EstructuraVis.DadesDirectX.D3DDeviceV3.CreateVertexBuffer _
(Len(m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)) * 2, 0, TL_FVF, D3DPOOL_DEFAULT)
D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferBarraAuxiliarV3, 0, _
Len(m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)) * 2, 0,
m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)

'Set EstructuraVis.DadesDirectX.VBufferBarraAuxiliarV4 =
EstructuraVis.DadesDirectX.D3DDeviceV4.CreateVertexBuffer _
(Len(m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)) * 2, 0, TL_FVF, D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferBarraAuxiliarV4, 0, _
Len(m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)) * 2, 0,
m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)

End Sub
```

```
Sub AssignaGeometriaLiniaAuxiliarIntroduccioBarresII(X1 As Double, Y1 As Double, Z1 As
Double, _
X2 As Double, Y2 As Double, Z2 As Double)

m_VertexsDeLiniaAuxiliarIntroduccioBarres(1) = CreateLitVertex((X1), (Y1), (Z1),
QBColor(12), 1, 0, 0)

m_VertexsDeLiniaAuxiliarIntroduccioBarres(2) = CreateLitVertex((X2), (Y2), (Z2),
QBColor(12), 1, 0, 0)

'Set EstructuraVis.DadesDirectX.VBufferBarraAuxiliarV1 =
EstructuraVis.DadesDirectX.D3DDeviceV1.CreateVertexBuffer _
(Len(m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)) * 2, 0, TL_FVF, D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferBarraAuxiliarV1, 0, _
Len(m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)) * 2, 0,
m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)

'Set EstructuraVis.DadesDirectX.VBufferBarraAuxiliarV2 =
EstructuraVis.DadesDirectX.D3DDeviceV2.CreateVertexBuffer _
(Len(m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)) * 2, 0, TL_FVF, D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferBarraAuxiliarV2, 0, _
Len(m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)) * 2, 0,
m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)

Set EstructuraVis.DadesDirectX.VBufferBarraAuxiliarV3 =
EstructuraVis.DadesDirectX.D3DDeviceV3.CreateVertexBuffer _
(Len(m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)) * 2, 0, TL_FVF, D3DPOOL_DEFAULT)
D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferBarraAuxiliarV3, 0, _
Len(m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)) * 2, 0,
m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)

'Set EstructuraVis.DadesDirectX.VBufferBarraAuxiliarV4 =
EstructuraVis.DadesDirectX.D3DDeviceV4.CreateVertexBuffer _
(Len(m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)) * 2, 0, TL_FVF, D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferBarraAuxiliarV4, 0, _
Len(m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)) * 2, 0,
m_VertexsDeLiniaAuxiliarIntroduccioBarres(1)

EstructuraVis.AuxFormularis.NodeInicialLiniaAuxiliar = 0
EstructuraVis.AuxFormularis.BufferLiniaAuxiliarBarresEmplenat = True
End Sub


Sub AssignaGeometriaForcesPuntualsATotesLesVistes()
Dim NodeEnCurs As Long
Dim MaxNode As Long
Dim SumaDeTotsElsComponents As Double
Dim ValorMaximDeForça As Double
Dim Modul As Double
```

```
Dim AccioEnCurs As Long

Dim V1 As D3DVECTOR
Dim V2 As D3DVECTOR
Dim V3 As D3DVECTOR
Dim V4 As D3DVECTOR

Dim LdC As Double 'Longitud del cap
Dim GdC As Double 'Gruix del cap
Dim LTotal As Double 'Longitud del "pal" de la força

Dim Color1 As Long, Color2 As Long


'Ressetegem els Búffers


EstructuraVis.AuxFormularis.BufferForcesNodeEmplenat = False
If EstructuraVis.ColNode.Count = 0 Then Exit Sub
'If Principal.HiHaAlgunaForçaNodal = False Then Exit Sub

'Ressetegem els Búffers
Set EstructuraVis.DadesDirectX.VBufferForcesV1 = Nothing
Set EstructuraVis.DadesDirectX.VBufferForcesV2 = Nothing
Set EstructuraVis.DadesDirectX.VBufferForcesV3 = Nothing
Set EstructuraVis.DadesDirectX.VBufferForcesV4 = Nothing

Set EstructuraVis.DadesDirectX.VBufferCapsDeForcesV1 = Nothing
Set EstructuraVis.DadesDirectX.VBufferCapsDeForcesV2 = Nothing
Set EstructuraVis.DadesDirectX.VBufferCapsDeForcesV3 = Nothing
Set EstructuraVis.DadesDirectX.VBufferCapsDeForcesV4 = Nothing


  MaxNode = EstructuraVis.ColNode.Count
  'MaxAccio = EstructuraVis.ColAccio.Count
  AccioEnCurs = EstructuraVis.AuxFormularis.AccioActivada


  ReDim m_VertexsDeForces(1 To MaxNode * 2)
  ReDim m_VertexsDeCapsDeForces(1 To MaxNode * 48)

  LTotal = EstructuraVis.Preferencies.DimensioDeLaLongitudDeLaLiniaDeForces
  LdC = EstructuraVis.Preferencies.DimensioDeLaLongitudDeLaPiramideDeForces
  GdC = EstructuraVis.Preferencies.DimensioSiseDeLaBaseDeLaPiramideDeForces
  Color1 = EstructuraVis.Preferencies.ColorForces
  Color2 = EstructuraVis.Preferencies.ColorForcesCap
```

```
   ValorMaximDeForça = 0

   'DETERMINO LA FORÇA MÀXIMA FENT UN RECORREGUT PER TOTS ELS NODES

   For NodeEnCurs = 1 To MaxNode

      If EstructuraVis.ColNode.Item(NodeEnCurs).ExisteixLaAccioNodal(Str(AccioEnCurs))
= True Then


            With
EstructuraVis.ColNode.Item(NodeEnCurs).ColSolicitacioNodal.Item(Str(AccioEnCurs))

            SumaDeTotsElsComponents = (.Fx ^ 2 + .Fy ^ 2 + .Fz ^ 2) ^ 0.5

            If Abs(ValorMaximDeForça) < SumaDeTotsElsComponents Then
            ValorMaximDeForça = (.Fx ^ 2 + .Fy ^ 2 + .Fz ^ 2) ^ 0.5
            End If
            End With


      End If

   Next NodeEnCurs


   If ValorMaximDeForça = 0 Then ValorMaximDeForça = 1

   'FI DETERMINACIÓ DE LA FORÇA MÀXIMA

   LTotal = LTotal *
DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs
   LdC = LdC * DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs
   GdC = GdC * DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs




   'DIBUIXEM LES PIRÀMIDES DE BASE ORTOGONAL

   For NodeEnCurs = 1 To MaxNode


      If EstructuraVis.ColNode.Item(NodeEnCurs).ExisteixLaAccioNodal(Str(AccioEnCurs))
= True Then

      Else
      GoTo Etiqueta
      End If
```

```
        'ESTABLEIXO ELS COLORS
        Color1 =
D3DColorRGBA(EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(1), _
        EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(2), _
        EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(3), 255)

        'LI RESTO 50 PERQUÈ TINGUI UNA TONALITAT DIFERENT
        Color2 =
D3DColorRGBA(EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(1) + 20, _
        EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(2) + 20, _
        EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(3) + 20, 255)


        If
EstructuraVis.ColNode.Item(NodeEnCurs).ColSolicitacioNodal.Item(Str(AccioEnCurs)).Forç
aActivada = True Then
        Color1 = EstructuraVis.Preferencies.Color1ForcesSeleccionades
        Color2 = EstructuraVis.Preferencies.Color2ForcesSeleccionades
        End If

        'FI ESTABLIMENT COLORS


          With
EstructuraVis.ColNode.Item(NodeEnCurs).ColSolicitacioNodal.Item(Str(AccioEnCurs))



            m_VertexsDeForces((NodeEnCurs - 1) * 2 + 1) = _
            CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X, _
            EstructuraVis.ColNode.Item(NodeEnCurs).Y, _
            EstructuraVis.ColNode.Item(NodeEnCurs).Z, Color1, Color1, 0, 0)

            m_VertexsDeForces((NodeEnCurs - 1) * 2 + 2) = CreateLitVertex( _
            EstructuraVis.ColNode.Item(NodeEnCurs).X - .Fx / ValorMaximDeForça * LTotal,
_
            EstructuraVis.ColNode.Item(NodeEnCurs).Y - .Fy / ValorMaximDeForça * LTotal,
_
            EstructuraVis.ColNode.Item(NodeEnCurs).Z - .Fz / ValorMaximDeForça * LTotal,
Color1, Color1, 1, 1)



          V1 = CreaVector(.Fx, .Fy, .Fz)

            If D3DXVec3Length(V1) <> 0 Then
            V1.X = V1.X / (.Fx ^ 2 + .Fy ^ 2 + .Fz ^ 2) ^ 0.5
            V1.Y = V1.Y / (.Fx ^ 2 + .Fy ^ 2 + .Fz ^ 2) ^ 0.5
            V1.Z = V1.Z / (.Fx ^ 2 + .Fy ^ 2 + .Fz ^ 2) ^ 0.5
```

```
            End If

        V2 = DMdlDirectX.VectorNormalMirantCapAmuntINormalitzat(V1)
        V3 = DMdlDirectX.VectorOrtogonalMirantCapAlCostatINormalitzat(V1)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 1) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X,
EstructuraVis.ColNode.Item(NodeEnCurs).Y, EstructuraVis.ColNode.Item(NodeEnCurs).Z,
Color1, Color1, 0, 0)
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 2) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC + V2.X * 3 * GdC
+ V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC + V2.Y * 3 * GdC + V3.Y * GdC,
_

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC + V2.Z * 3 * GdC + V3.Z * GdC,
Color1, Color1, 0, 0)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 3) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC + V2.X * GdC +
V3.X * 3 * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC + V2.Y * GdC + V3.Y * 3 * GdC,
_

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC + V2.Z * GdC + V3.Z * 3 * GdC,
Color1, Color1, 0, 0)



        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 4) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X,
EstructuraVis.ColNode.Item(NodeEnCurs).Y, EstructuraVis.ColNode.Item(NodeEnCurs).Z,
Color1, Color1, 0, 0)
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 5) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC + V2.X * GdC +
V3.X * 3 * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC + V2.Y * GdC + V3.Y * 3 * GdC,
_

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC + V2.Z * GdC + V3.Z * 3 * GdC,
Color2, Color2, 0, 0)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 6) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC - V2.X * GdC +
V3.X * 3 * GdC, _
```

```
EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC - V2.Y * GdC + V3.Y * 3 * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC - V2.Z * GdC + V3.Z * 3 * GdC,
Color2, Color2, 0, 0)


        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 7) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X,
EstructuraVis.ColNode.Item(NodeEnCurs).Y, EstructuraVis.ColNode.Item(NodeEnCurs).Z,
Color1, Color1, 0, 0)
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 8) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC - V2.X * GdC +
V3.X * 3 * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC - V2.Y * GdC + V3.Y * 3 * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC - V2.Z * GdC + V3.Z * 3 * GdC,
Color1, Color1, 0, 0)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 9) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC - V2.X * 3 * GdC
+ V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC - V2.Y * 3 * GdC + V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC - V2.Z * 3 * GdC + V3.Z * GdC,
Color1, Color1, 0, 0)


        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 10) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X,
EstructuraVis.ColNode.Item(NodeEnCurs).Y, EstructuraVis.ColNode.Item(NodeEnCurs).Z,
Color1, Color1, 0, 0)
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 11) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC - V2.X * 3 * GdC
+ V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC - V2.Y * 3 * GdC + V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC - V2.Z * 3 * GdC + V3.Z * GdC,
Color2, Color2, 0, 0)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 12) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC - V2.X * 3 * GdC
- V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC - V2.Y * 3 * GdC - V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC - V2.Z * 3 * GdC - V3.Z * GdC,
Color2, Color2, 0, 0)
```

```
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 13) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X,
EstructuraVis.ColNode.Item(NodeEnCurs).Y, EstructuraVis.ColNode.Item(NodeEnCurs).Z,
Color1, Color1, 0, 0)
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 14) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC - V2.X * 3 * GdC
- V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC - V2.Y * 3 * GdC - V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC - V2.Z * 3 * GdC - V3.Z * GdC,
Color1, Color1, 0, 0)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 15) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC - V2.X * GdC - 3
* V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC - V2.Y * GdC - 3 * V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC - V2.Z * GdC - 3 * V3.Z * GdC,
Color1, Color1, 0, 0)


        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 16) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X,
EstructuraVis.ColNode.Item(NodeEnCurs).Y, EstructuraVis.ColNode.Item(NodeEnCurs).Z,
Color1, Color1, 0, 0)
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 17) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC - V2.X * GdC - 3
* V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC - V2.Y * GdC - 3 * V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC - V2.Z * GdC - 3 * V3.Z * GdC,
Color2, Color2, 0, 0)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 18) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC + V2.X * GdC - 3
* V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC + V2.Y * GdC - 3 * V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC + V2.Z * GdC - 3 * V3.Z * GdC,
Color2, Color2, 0, 0)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 19) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X,
EstructuraVis.ColNode.Item(NodeEnCurs).Y, EstructuraVis.ColNode.Item(NodeEnCurs).Z,
Color1, Color1, 0, 0)
```

```
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 20) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC + V2.X * 3 * GdC
- V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC + V2.Y * 3 * GdC - V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC + V2.Z * 3 * GdC - V3.Z * GdC,
Color1, Color1, 0, 0)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 21) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC + V2.X * GdC - 3
* V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC + V2.Y * GdC - 3 * V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC + V2.Z * GdC - 3 * V3.Z * GdC,
Color1, Color1, 0, 0)


        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 22) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X,
EstructuraVis.ColNode.Item(NodeEnCurs).Y, EstructuraVis.ColNode.Item(NodeEnCurs).Z,
Color1, Color1, 0, 0)
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 23) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC + V2.X * 3 * GdC
- V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC + V2.Y * 3 * GdC - V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC + V2.Z * 3 * GdC - V3.Z * GdC,
Color2, Color2, 0, 0)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 24) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC + V2.X * 3 * GdC
+ V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC + V2.Y * 3 * GdC + V3.Y * GdC,
_

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC + V2.Z * 3 * GdC + V3.Z * GdC,
Color2, Color2, 0, 0)


        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 25) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC,
EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC,
EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC, Color2, Color2, 0, 0)
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 26) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC + V2.X * 3 * GdC
+ V3.X * GdC, _
```

```
EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC + V2.Y * 3 * GdC + V3.Y * GdC,
_

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC + V2.Z * 3 * GdC + V3.Z * GdC,
Color2, Color2, 0, 0)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 27) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC + V2.X * GdC +
V3.X * 3 * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC + V2.Y * GdC + V3.Y * 3 * GdC,
_

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC + V2.Z * GdC + V3.Z * 3 * GdC,
Color2, Color2, 0, 0)



        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 28) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC,
EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC,
EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC, Color1, Color1, 0, 0)
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 29) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC + V2.X * GdC +
V3.X * 3 * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC + V2.Y * GdC + V3.Y * 3 * GdC,
_

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC + V2.Z * GdC + V3.Z * 3 * GdC,
Color1, Color1, 0, 0)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 30) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC - V2.X * GdC +
V3.X * 3 * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC - V2.Y * GdC + V3.Y * 3 * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC - V2.Z * GdC + V3.Z * 3 * GdC,
Color1, Color1, 0, 0)



        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 31) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC,
EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC,
EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC, Color2, Color2, 0, 0)
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 32) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC - V2.X * GdC +
V3.X * 3 * GdC, _
```

```
EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC - V2.Y * GdC + V3.Y * 3 * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC - V2.Z * GdC + V3.Z * 3 * GdC,
Color2, Color2, 0, 0)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 33) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC - V2.X * 3 * GdC
+ V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC - V2.Y * 3 * GdC + V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC - V2.Z * 3 * GdC + V3.Z * GdC,
Color2, Color2, 0, 0)


        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 34) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC,
EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC,
EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC, Color1, Color1, 0, 0)
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 35) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC - V2.X * 3 * GdC
+ V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC - V2.Y * 3 * GdC + V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC - V2.Z * 3 * GdC + V3.Z * GdC,
Color1, Color1, 0, 0)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 36) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC - V2.X * 3 * GdC
- V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC - V2.Y * 3 * GdC - V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC - V2.Z * 3 * GdC - V3.Z * GdC,
Color1, Color1, 0, 0)


        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 37) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC,
EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC,
EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC, Color2, Color2, 0, 0)
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 38) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC - V2.X * 3 * GdC
- V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC - V2.Y * 3 * GdC - V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC - V2.Z * 3 * GdC - V3.Z * GdC,
Color2, Color2, 0, 0)
```

```
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 39) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC - V2.X * GdC - 3
* V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC - V2.Y * GdC - 3 * V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC - V2.Z * GdC - 3 * V3.Z * GdC,
Color2, Color2, 0, 0)


        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 40) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC,
EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC,
EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC, Color1, Color1, 0, 0)
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 41) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC - V2.X * GdC - 3
* V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC - V2.Y * GdC - 3 * V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC - V2.Z * GdC - 3 * V3.Z * GdC,
Color1, Color1, 0, 0)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 42) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC + V2.X * GdC - 3
* V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC + V2.Y * GdC - 3 * V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC + V2.Z * GdC - 3 * V3.Z * GdC,
Color1, Color1, 0, 0)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 43) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC,
EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC,
EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC, Color2, Color2, 0, 0)
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 44) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC + V2.X * 3 * GdC
- V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC + V2.Y * 3 * GdC - V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC + V2.Z * 3 * GdC - V3.Z * GdC,
Color2, Color2, 0, 0)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 45) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC + V2.X * GdC - 3
* V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC + V2.Y * GdC - 3 * V3.Y * GdC, _
```

```
EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC + V2.Z * GdC - 3 * V3.Z * GdC,
Color2, Color2, 0, 0)


        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 46) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC,
EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC,
EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC, Color1, Color1, 0, 0)
        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 47) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC + V2.X * 3 * GdC
- V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC + V2.Y * 3 * GdC - V3.Y * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC + V2.Z * 3 * GdC - V3.Z * GdC,
Color1, Color1, 0, 0)

        m_VertexsDeCapsDeForces((NodeEnCurs - 1) * 48 + 48) =
CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X - V1.X * LdC + V2.X * 3 * GdC
+ V3.X * GdC, _

EstructuraVis.ColNode.Item(NodeEnCurs).Y - V1.Y * LdC + V2.Y * 3 * GdC + V3.Y * GdC,
_

EstructuraVis.ColNode.Item(NodeEnCurs).Z - V1.Z * LdC + V2.Z * 3 * GdC + V3.Z * GdC,
Color1, Color1, 0, 0)



        End With
Etiqueta:

    Next NodeEnCurs

Etiqueta2:



'Buffers Corresponents a les forces
'Set EstructuraVis.DadesDirectX.VBufferForcesV1 =
EstructuraVis.DadesDirectX.D3DDeviceV1.CreateVertexBuffer _
(Len(m_VertexsDeForces(1)) * EstructuraVis.ColNode.Count * 2, 0, TL_FVF,
D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferForcesV1, 0, _
Len(m_VertexsDeForces(1)) * EstructuraVis.ColNode.Count * 2, 0, m_VertexsDeForces(1)

'Set EstructuraVis.DadesDirectX.VBufferForcesV2 =
EstructuraVis.DadesDirectX.D3DDeviceV2.CreateVertexBuffer _
(Len(m_VertexsDeForces(1)) * EstructuraVis.ColNode.Count * 2, 0, TL_FVF,
D3DPOOL_DEFAULT)
```

```
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferForcesV2, 0, _
Len(m_VertexsDeForces(1)) * EstructuraVis.ColNode.Count * 2, 0, m_VertexsDeForces(1)

Set EstructuraVis.DadesDirectX.VBufferForcesV3 =
EstructuraVis.DadesDirectX.D3DDeviceV3.CreateVertexBuffer _
(Len(m_VertexsDeForces(1)) * EstructuraVis.ColNode.Count * 2, 0, TL_FVF,
D3DPOOL_DEFAULT)
D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferForcesV3, 0, _
Len(m_VertexsDeForces(1)) * EstructuraVis.ColNode.Count * 2, 0, m_VertexsDeForces(1)

'Set EstructuraVis.DadesDirectX.VBufferForcesV4 =
EstructuraVis.DadesDirectX.D3DDeviceV4.CreateVertexBuffer _
(Len(m_VertexsDeForces(1)) * EstructuraVis.ColNode.Count * 2, 0, TL_FVF,
D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferForcesV4, 0, _
Len(m_VertexsDeForces(1)) * EstructuraVis.ColNode.Count * 2, 0, m_VertexsDeForces(1)

'Buffers Corresponents als caps de les forces

'Set EstructuraVis.DadesDirectX.VBufferCapsDeForcesV1 =
EstructuraVis.DadesDirectX.D3DDeviceV1.CreateVertexBuffer _
(Len(m_VertexsDeCapsDeForces(1)) * EstructuraVis.ColNode.Count * 48, 0, TL_FVF,
D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferCapsDeForcesV1, 0, _
Len(m_VertexsDeCapsDeForces(1)) * EstructuraVis.ColNode.Count * 48, 0,
m_VertexsDeCapsDeForces(1)

'Set EstructuraVis.DadesDirectX.VBufferCapsDeForcesV2 =
EstructuraVis.DadesDirectX.D3DDeviceV2.CreateVertexBuffer _
(Len(m_VertexsDeCapsDeForces(1)) * EstructuraVis.ColNode.Count * 48, 0, TL_FVF,
D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferCapsDeForcesV2, 0, _
Len(m_VertexsDeCapsDeForces(1)) * EstructuraVis.ColNode.Count * 48, 0,
m_VertexsDeCapsDeForces(1)

Set EstructuraVis.DadesDirectX.VBufferCapsDeForcesV3 =
EstructuraVis.DadesDirectX.D3DDeviceV3.CreateVertexBuffer _
(Len(m_VertexsDeCapsDeForces(1)) * EstructuraVis.ColNode.Count * 48, 0, TL_FVF,
D3DPOOL_DEFAULT)
D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferCapsDeForcesV3, 0, _
Len(m_VertexsDeCapsDeForces(1)) * EstructuraVis.ColNode.Count * 48, 0,
m_VertexsDeCapsDeForces(1)

'Set EstructuraVis.DadesDirectX.VBufferCapsDeForcesV4 =
EstructuraVis.DadesDirectX.D3DDeviceV4.CreateVertexBuffer _
(Len(m_VertexsDeCapsDeForces(1)) * EstructuraVis.ColNode.Count * 48, 0, TL_FVF,
D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferCapsDeForcesV4, 0, _
Len(m_VertexsDeCapsDeForces(1)) * EstructuraVis.ColNode.Count * 48, 0,
m_VertexsDeCapsDeForces(1)
```

```
EstructuraVis.AuxFormularis.BufferForcesNodeEmplenat = True
End Sub



Sub AssignaGeometriaMomentsNodalsATotesLesVistes()
Dim NodeEnCurs As Long
Dim MaxNode As Long
Dim SumaDeTotsElsComponents As Double
Dim ValorMaximDeForça As Double
Dim Modul As Double
Dim i As Integer
Dim AccioEnCurs As Long

Dim VD As D3DVECTOR
Dim VN As D3DVECTOR
Dim VC As D3DVECTOR

Dim LdC As Double 'Longitud del cap
Dim GdC As Double 'Gruix del cap
Dim LTotal As Double 'Longitud del "pal" de la força

Dim Color1 As Long, Color2 As Long, Color3 As Long, Color4 As Long
Dim P(1 To 12) As D3DVECTOR



If EstructuraVis.ColNode.Count = 0 Then
EstructuraVis.AuxFormularis.BufferMomentsEmplenat = False
Exit Sub
End If

    MaxNode = EstructuraVis.ColNode.Count
   'MaxAccio = EstructuraVis.ColAccio.Count
   AccioEnCurs = EstructuraVis.AuxFormularis.AccioActivada


   ReDim m_VertexsDeMomentsNode(1 To MaxNode * 2)
   ReDim m_VertexsDeCapsDeMomentsNode(1 To MaxNode * 24)


'Ressetegem els Búffers
If Principal.HiHaAlgunMomentNodal = False Then
Set EstructuraVis.DadesDirectX.VBufferMomentsNode = Nothing
Set EstructuraVis.DadesDirectX.VBufferCapsMomentsNode = Nothing
EstructuraVis.AuxFormularis.BufferMomentsEmplenat = False
GoTo Etiqueta2
End If
```

```
  LTotal = EstructuraVis.Preferencies.DimensioDeLaLongitudDeLaLiniaDeForces
  LdC = EstructuraVis.Preferencies.DimensioDeLaLongitudDeLaPiramideDeForces
  'GdC = EstructuraVis.Preferencies.DimensioSiseDeLaBaseDeLaPiramideDeForces

  'Color1 = EstructuraVis.Preferencies.ColorForces
  'Color2 = EstructuraVis.Preferencies.ColorForcesCap


   ValorMaximDeForça = 0

   'DETERMINO LA FORÇA MÀXIMA FENT UN RECORREGUT PER TOTS ELS NODES

   For NodeEnCurs = 1 To MaxNode

      If EstructuraVis.ColNode.Item(NodeEnCurs).ExisteixLaAccioNodal(Str(AccioEnCurs))
= True Then


            With
EstructuraVis.ColNode.Item(NodeEnCurs).ColSolicitacioNodal.Item(Str(AccioEnCurs))

            SumaDeTotsElsComponents = (.Mx ^ 2 + .My ^ 2 + .Mz ^ 2) ^ 0.5

            If Abs(ValorMaximDeForça) < SumaDeTotsElsComponents Then
            ValorMaximDeForça = (.Mx ^ 2 + .My ^ 2 + .Mz ^ 2) ^ 0.5
            End If
            End With


      End If

   Next NodeEnCurs


   If ValorMaximDeForça = 0 Then ValorMaximDeForça = 1

   'FI DETERMINACIÓ DE LA FORÇA MÀXIMA

   LTotal = LTotal *
DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs
   LdC = LdC * DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs
   'GdC = GdC * DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs




   'DIBUIXEM LES PIRÀMIDES DE BASE ORTOGONAL

   For NodeEnCurs = 1 To MaxNode
```

```
        If EstructuraVis.ColNode.Item(NodeEnCurs).ExisteixLaAccioNodal(Str(AccioEnCurs))
= True Then

        Else
        GoTo Etiqueta
        End If


        'ESTABLEIXO ELS COLORS
        Color1 =
D3DColorRGBA(EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(1), _
        EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(2), _
        EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(3), 255)

        'PERQUÈ TINGUI UNA TONALITAT DIFERENT
        Color2 =
D3DColorRGBA(EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(1) * 0.85,
_
        EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(2) * 0.85, _
        EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(3) * 0.85, 255)


        'PERQUÈ TINGUI UNA TONALITAT DIFERENT
        Color3 =
D3DColorRGBA(EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(1) * 0.7, _
        EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(2) * 0.7, _
        EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(3) * 0.7, 255)

        'PERQUÈ TINGUI UNA TONALITAT DIFERENT
        Color4 =
D3DColorRGBA(EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(1) * 0.55,
_
        EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(2) * 0.55, _
        EstructuraVis.ColAccio.Item(Str(AccioEnCurs)).ColorRGBAccio(3) * 0.55, 255)


        If
EstructuraVis.ColNode.Item(NodeEnCurs).ColSolicitacioNodal.Item(Str(AccioEnCurs)).Forç
aActivada = True Then
        Color1 = EstructuraVis.Preferencies.Color1ForcesSeleccionades
        Color2 = EstructuraVis.Preferencies.Color2ForcesSeleccionades
        Color2 = EstructuraVis.Preferencies.Color2ForcesSeleccionades
        End If

        'FI ESTABLIMENT COLORS


            With
EstructuraVis.ColNode.Item(NodeEnCurs).ColSolicitacioNodal.Item(Str(AccioEnCurs))
```

```
            m_VertexsDeMomentsNode((NodeEnCurs - 1) * 2 + 1) = _
            CreateLitVertex(EstructuraVis.ColNode.Item(NodeEnCurs).X, _
            EstructuraVis.ColNode.Item(NodeEnCurs).Y, _
            EstructuraVis.ColNode.Item(NodeEnCurs).Z, Color1, Color1, 0, 0)

            m_VertexsDeMomentsNode((NodeEnCurs - 1) * 2 + 2) = CreateLitVertex( _
            EstructuraVis.ColNode.Item(NodeEnCurs).X - .Mx / ValorMaximDeForça * LTotal,
_
            EstructuraVis.ColNode.Item(NodeEnCurs).Y - .My / ValorMaximDeForça * LTotal,
_
            EstructuraVis.ColNode.Item(NodeEnCurs).Z - .Mz / ValorMaximDeForça * LTotal,
Color1, Color1, 1, 1)



        VD = CreaVector(.Mx, .My, .Mz)

            If D3DXVec3Length(VD) <> 0 Then
            VD.X = VD.X / (.Mx ^ 2 + .My ^ 2 + .Mz ^ 2) ^ 0.5
            VD.Y = VD.Y / (.Mx ^ 2 + .My ^ 2 + .Mz ^ 2) ^ 0.5
            VD.Z = VD.Z / (.Mx ^ 2 + .My ^ 2 + .Mz ^ 2) ^ 0.5
            End If


        VN = DMdlDirectX.VectorNormalMirantCapAmuntINormalitzat(VD)
        VC = DMdlDirectX.VectorOrtogonalMirantCapAlCostatINormalitzat(VD)


        P(1) = CreaVector(EstructuraVis.ColNode.Item(NodeEnCurs).X,
EstructuraVis.ColNode.Item(NodeEnCurs).Y, _
        EstructuraVis.ColNode.Item(NodeEnCurs).Z)
        P(2) = CreaVector(P(1).X - VD.X * LdC + VN.X * LdC / 3, P(1).Y - VD.Y * LdC _
        + VN.Y * LdC / 3, P(1).Z - VD.Z * LdC + VN.Z * LdC / 3)
        P(3) = CreaVector(P(1).X - VD.X * LdC - VN.X * LdC / 3, P(1).Y - VD.Y * LdC _
        - VN.Y * LdC / 3, P(1).Z - VD.Z * LdC - VN.Z * LdC / 3)

        P(4) = CreaVector(EstructuraVis.ColNode.Item(NodeEnCurs).X - VD.X * LdC * 2, _
        EstructuraVis.ColNode.Item(NodeEnCurs).Y - VD.Y * LdC * 2, _
        EstructuraVis.ColNode.Item(NodeEnCurs).Z - VD.Z * LdC * 2)
        P(5) = CreaVector(P(4).X - VD.X * LdC + (VN.X) * LdC / 3, _
        P(4).Y - VD.Y * LdC + (VN.Y) * LdC / 3, P(4).Z - VD.Z * LdC + _
        (VN.Z) * LdC / 3)
        P(6) = CreaVector(P(4).X - VD.X * LdC + (-VN.X) * LdC / 3, _
        P(4).Y - VD.Y * LdC + (-VN.Y) * LdC / 3, P(4).Z - VD.Z * LdC + _
        (-VN.Z) * LdC / 3)


        P(7) = CreaVector(EstructuraVis.ColNode.Item(NodeEnCurs).X - VD.X * LdC, _
        EstructuraVis.ColNode.Item(NodeEnCurs).Y - VD.Y * LdC, _
```

```
        EstructuraVis.ColNode.Item(NodeEnCurs).Z - VD.Z * LdC)
        P(8) = CreaVector(P(7).X - VD.X * LdC + (VC.X) * LdC / 3, _
        P(7).Y - VD.Y * LdC + (VC.Y) * LdC / 3, P(7).Z - VD.Z * LdC + _
        (VC.Z) * LdC / 3)
        P(9) = CreaVector(P(7).X - VD.X * LdC + (-VC.X) * LdC / 3, _
        P(7).Y - VD.Y * LdC + (-VC.Y) * LdC / 3, P(7).Z - VD.Z * LdC + _
        (-VC.Z) * LdC / 3)



        P(10) = CreaVector(EstructuraVis.ColNode.Item(NodeEnCurs).X - VD.X * LdC * 2, _
        EstructuraVis.ColNode.Item(NodeEnCurs).Y - VD.Y * LdC * 2, _
        EstructuraVis.ColNode.Item(NodeEnCurs).Z - VD.Z * LdC * 2)
        P(11) = CreaVector(P(4).X - VD.X * LdC + (VC.X) * LdC / 3, _
        P(4).Y - VD.Y * LdC + (VC.Y) * LdC / 3, P(4).Z - VD.Z * LdC + _
        (VC.Z) * LdC / 3)
        P(12) = CreaVector(P(4).X - VD.X * LdC + (-VC.X) * LdC / 3, _
        P(4).Y - VD.Y * LdC + (-VC.Y) * LdC / 3, P(4).Z - VD.Z * LdC + _
        (-VC.Z) * LdC / 3)



        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 1) =
CreateLitVertex(P(1).X, P(1).Y, P(1).Z, Color1, Color1, 1, 1)
        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 2) =
CreateLitVertex(P(2).X, P(2).Y, P(2).Z, Color1, Color1, 1, 1)

        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 3) =
CreateLitVertex(P(2).X, P(2).Y, P(2).Z, Color1, Color1, 1, 1)
        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 4) =
CreateLitVertex(P(3).X, P(3).Y, P(3).Z, Color1, Color1, 1, 1)

        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 5) =
CreateLitVertex(P(3).X, P(3).Y, P(3).Z, Color1, Color1, 1, 1)
        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 6) =
CreateLitVertex(P(1).X, P(1).Y, P(1).Z, Color1, Color1, 1, 1)

        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 7) =
CreateLitVertex(P(4).X, P(4).Y, P(4).Z, Color2, Color2, 1, 1)
        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 8) =
CreateLitVertex(P(5).X, P(5).Y, P(5).Z, Color2, Color2, 1, 1)

        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 9) =
CreateLitVertex(P(5).X, P(5).Y, P(5).Z, Color2, Color2, 1, 1)
        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 10) =
CreateLitVertex(P(6).X, P(6).Y, P(6).Z, Color2, Color2, 1, 1)

        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 11) =
CreateLitVertex(P(6).X, P(6).Y, P(6).Z, Color2, Color2, 1, 1)
        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 12) =
CreateLitVertex(P(4).X, P(4).Y, P(4).Z, Color2, Color2, 1, 1)
```

```
        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 13) =
CreateLitVertex(P(7).X, P(7).Y, P(7).Z, Color3, Color3, 1, 1)
        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 14) =
CreateLitVertex(P(8).X, P(8).Y, P(8).Z, Color3, Color3, 1, 1)

        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 15) =
CreateLitVertex(P(8).X, P(8).Y, P(8).Z, Color3, Color3, 1, 1)
        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 16) =
CreateLitVertex(P(9).X, P(9).Y, P(9).Z, Color3, Color3, 1, 1)

        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 17) =
CreateLitVertex(P(9).X, P(9).Y, P(9).Z, Color3, Color3, 1, 1)
        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 18) =
CreateLitVertex(P(7).X, P(7).Y, P(7).Z, Color3, Color3, 1, 1)

        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 19) =
CreateLitVertex(P(10).X, P(10).Y, P(10).Z, Color4, Color4, 1, 1)
        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 20) =
CreateLitVertex(P(11).X, P(11).Y, P(11).Z, Color4, Color4, 1, 1)

        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 21) =
CreateLitVertex(P(11).X, P(11).Y, P(11).Z, Color4, Color4, 1, 1)
        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 22) =
CreateLitVertex(P(12).X, P(12).Y, P(12).Z, Color4, Color4, 1, 1)

        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 23) =
CreateLitVertex(P(12).X, P(12).Y, P(12).Z, Color4, Color4, 1, 1)
        m_VertexsDeCapsDeMomentsNode((NodeEnCurs - 1) * 24 + 24) =
CreateLitVertex(P(10).X, P(10).Y, P(10).Z, Color4, Color4, 1, 1)


        End With
Etiqueta:

   Next NodeEnCurs

Etiqueta2:




Set EstructuraVis.DadesDirectX.VBufferMomentsNode =
EstructuraVis.DadesDirectX.D3DDeviceV3.CreateVertexBuffer _
(Len(m_VertexsDeMomentsNode(1)) * EstructuraVis.ColNode.Count * 2, 0, TL_FVF,
D3DPOOL_DEFAULT)
D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferMomentsNode, 0, _
Len(m_VertexsDeMomentsNode(1)) * EstructuraVis.ColNode.Count * 2, 0,
m_VertexsDeMomentsNode(1)
```

```
Set EstructuraVis.DadesDirectX.VBufferCapsMomentsNode =
EstructuraVis.DadesDirectX.D3DDeviceV3.CreateVertexBuffer _
(Len(m_VertexsDeCapsDeMomentsNode(1)) * EstructuraVis.ColNode.Count * 24, 0,
TL_FVF, D3DPOOL_DEFAULT)
D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferCapsMomentsNode, 0, _
Len(m_VertexsDeCapsDeMomentsNode(1)) * EstructuraVis.ColNode.Count * 24, 0,
m_VertexsDeCapsDeMomentsNode(1)


EstructuraVis.AuxFormularis.BufferMomentsEmplenat = True
End Sub



Sub AssignaGeometriaForcesBarraATotesLesVistes()
Dim BarraEnCurs As Long
Dim MaxBarra As Long
Dim ForçaEnCurs As Long
Dim ForçaEnCursAbsoluta As Long
Dim MaxForça As Long
Dim NumForces As Long


Dim SumaDeTotsElsComponents As Double
Dim ValorMaximDeForça As Double
Dim Modul As Double

Dim AccioEnCurs As Long

Dim V1 As D3DVECTOR
Dim V2 As D3DVECTOR
Dim V3 As D3DVECTOR
Dim V4 As D3DVECTOR

Dim VDirector As D3DVECTOR
Dim VNormal As D3DVECTOR
Dim VCostat As D3DVECTOR
Dim VAux1 As D3DVECTOR
Dim VAux2 As D3DVECTOR

Dim LdC As Double 'Longitud del cap
Dim GdC As Double 'Gruix del cap
Dim LTotal As Double 'Longitud del "pal" de la força

Dim Color1 As Long, Color2 As Long

Dim X1 As Double, Y1 As Double, Z1 As Double
Dim Fx As Double, Fy As Double, Fz As Double
Dim Zini As Double
```

```
'Ressetegem els Búffers
Set EstructuraVis.DadesDirectX.VBufferForcesBarraV1 = Nothing
Set EstructuraVis.DadesDirectX.VBufferForcesBarraV2 = Nothing
Set EstructuraVis.DadesDirectX.VBufferForcesBarraV3 = Nothing
Set EstructuraVis.DadesDirectX.VBufferForcesBarraV4 = Nothing

Set EstructuraVis.DadesDirectX.VBufferCapsDeForcesBarraV1 = Nothing
Set EstructuraVis.DadesDirectX.VBufferCapsDeForcesBarraV2 = Nothing
Set EstructuraVis.DadesDirectX.VBufferCapsDeForcesBarraV3 = Nothing
Set EstructuraVis.DadesDirectX.VBufferCapsDeForcesBarraV4 = Nothing


  MaxBarra = EstructuraVis.ColBarra.Count
  AccioEnCurs = EstructuraVis.AuxFormularis.AccioActivada



  'CALCULO EL NOMBRE DE FORCES BARRA QUE HI HA
  NumForces = 0
  For BarraEnCurs = 1 To MaxBarra
     If
EstructuraVis.ColBarra.Item(BarraEnCurs).ExisteixLaColeccioDeForcesNodalsAgrupadesP
erAccio(AccioEnCurs) = True Then

       NumForces = NumForces +
EstructuraVis.ColBarra.Item(BarraEnCurs).ColForcesPuntualsAgrupadesPerAccio. _
       Item(Str(AccioEnCurs)).ColCarreguesPuntuals.Count
       End If
  Next BarraEnCurs
  'FI CÀLCUL; DADA EMMAGATZEMADA A NumForces
  EstructuraVis.AuxFormularis.NumeroDeForcesEnBarra = NumForces

  If NumForces = 0 Then Exit Sub

  ReDim m_VertexsDeForcesBarra(1 To NumForces * 2)
  ReDim m_VertexsDeCapsDeForcesBarra(1 To NumForces * 48)

  LTotal = EstructuraVis.Preferencies.DimensioDeLaLongitudDeLaLiniaDeForces
  LdC = EstructuraVis.Preferencies.DimensioDeLaLongitudDeLaPiramideDeForces
  GdC = EstructuraVis.Preferencies.DimensioSiseDeLaBaseDeLaPiramideDeForces

   ValorMaximDeForça = 0

   'DETERMINO LA FORÇA MÀXIMA FENT UN RECORREGUT PER TOTES LES
BARRES

   For BarraEnCurs = 1 To MaxBarra

      If
EstructuraVis.ColBarra.Item(BarraEnCurs).ExisteixLaColeccioDeForcesNodalsAgrupadesP
erAccio(AccioEnCurs) = True Then
```

```
        If
EstructuraVis.ColBarra.Item(BarraEnCurs).ColForcesPuntualsAgrupadesPerAccio.Item(Str(
AccioEnCurs)) _
        .ModulForçaMaximaDeTotaLaColeccioDeForcesPuntualsEnBarra >
ValorMaximDeForça Then

          ValorMaximDeForça = EstructuraVis.ColBarra.Item(BarraEnCurs). _
          ColForcesPuntualsAgrupadesPerAccio.Item(Str(AccioEnCurs)) _
          .ModulForçaMaximaDeTotaLaColeccioDeForcesPuntualsEnBarra

        End If

      End If
    Next BarraEnCurs


    If ValorMaximDeForça = 0 Then ValorMaximDeForça = 1

    'FI DETERMINACIÓ DE LA FORÇA MÀXIMA

    LTotal = LTotal *
DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs
    LdC = LdC * DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs
    GdC = GdC * DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs



  ForçaEnCursAbsoluta = 0

    'DIBUIXEM LES PIRÀMIDES DE BASE ORTOGONAL

    For BarraEnCurs = 1 To MaxBarra




        If
EstructuraVis.ColBarra.Item(BarraEnCurs).ExisteixLaColeccioDeForcesNodalsAgrupadesP
erAccio(AccioEnCurs) = True Then

        MaxForça =
EstructuraVis.ColBarra.Item(BarraEnCurs).ColForcesPuntualsAgrupadesPerAccio _
        .Item(Str(AccioEnCurs)).ColCarreguesPuntuals.Count
        Else
        MaxForça = 0
        End If



        For ForçaEnCurs = 1 To MaxForça
```

```
        ForçaEnCursAbsoluta = ForçaEnCursAbsoluta + 1


        'PROPIETATS GEOMÈTRIQUES DE LA BARRA
           With EstructuraVis.ColBarra.Item(BarraEnCurs)
           VDirector = TrobaVectorDirector(CreaVector(.X1, .Y1, .Z1), CreaVector(.X2, .Y2,
.Z2))
           D3DXVec3Normalize VDirector, VDirector

           VCostat = CreaVector(.VxOrto, .VyOrto, .VzOrto)
           D3DXVec3Normalize VCostat, VCostat

           D3DXVec3Cross VNormal, VDirector, VCostat
           D3DXVec3Normalize VNormal, VNormal

           X1 = .X1: Y1 = .Y1: Z1 = .Z1
           End With

        'FI PROPIETATS GEOMÈTRIQUES DE LA BARRA


        Color1 =
D3DColorRGBA(EstructuraVis.ColAccio.Item(AccioEnCurs).ColorRGBAccio(1), _
        EstructuraVis.ColAccio.Item(AccioEnCurs).ColorRGBAccio(2), _
        EstructuraVis.ColAccio.Item(AccioEnCurs).ColorRGBAccio(3), 255)

        'LI RESTO 50 PERQUÈ TINGUI UNA TONALITAT DIFERENT
        Color2 =
D3DColorRGBA(EstructuraVis.ColAccio.Item(AccioEnCurs).ColorRGBAccio(1) + 20, _
        EstructuraVis.ColAccio.Item(AccioEnCurs).ColorRGBAccio(2) + 20, _
        EstructuraVis.ColAccio.Item(AccioEnCurs).ColorRGBAccio(3) + 20, 255)




           'With
EstructuraVis.ColNode.Item(NodeEnCurs).ColSolicitacioNodal.Item(Str(AccioEnCurs))
           With
EstructuraVis.ColBarra.Item(BarraEnCurs).ColForcesPuntualsAgrupadesPerAccio _
           .Item(Str(AccioEnCurs)).ColCarreguesPuntuals.Item(ForçaEnCurs)

           Zini = .ZCarregaIni
           Fx = .Fx: Fy = .Fy: Fz = .Fz
```

```
            m_VertexsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 2 + 1) = _
            CreateLitVertex(X1 + Zini * VDirector.X, Y1 + Zini * VDirector.Y _
            , Z1 + Zini * VDirector.Z, Color1, Color1, 0, 0)

            m_VertexsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 2 + 2) = CreateLitVertex(
_
            X1 + Zini * VDirector.X - Fx / ValorMaximDeForça * LTotal, _
            Y1 + Zini * VDirector.Y - Fy / ValorMaximDeForça * LTotal, _
            Z1 + Zini * VDirector.Z - Fz / ValorMaximDeForça * LTotal, Color1, Color1, 1, 1)

         V1 = CreaVector(.Fx, .Fy, .Fz)

            If D3DXVec3Length(V1) <> 0 Then
            V1.X = V1.X / (.Fx ^ 2 + .Fy ^ 2 + .Fz ^ 2) ^ 0.5
            V1.Y = V1.Y / (.Fx ^ 2 + .Fy ^ 2 + .Fz ^ 2) ^ 0.5
            V1.Z = V1.Z / (.Fx ^ 2 + .Fy ^ 2 + .Fz ^ 2) ^ 0.5
            End If


         V2 = DMdlDirectX.VectorNormalMirantCapAmuntINormalitzat(V1)
         V3 = DMdlDirectX.VectorOrtogonalMirantCapAlCostatINormalitzat(V1)

         X1 = X1 + Zini * VDirector.X
         Y1 = Y1 + Zini * VDirector.Y
         Z1 = Z1 + Zini * VDirector.Z

         m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 1) =
CreateLitVertex(((X1)), (Y1), (Z1), Color1, Color1, 0, 0)
         m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 2) =
CreateLitVertex((X1) - V1.X * LdC + V2.X * 3 * GdC + V3.X * GdC, _
                                                (Y1) - V1.Y * LdC + V2.Y * 3 * GdC +
V3.Y * GdC, _
                                                (Z1) - V1.Z * LdC + V2.Z * 3 * GdC +
V3.Z * GdC, Color1, Color1, 0, 0)

         m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 3) =
CreateLitVertex((X1) - V1.X * LdC + V2.X * GdC + V3.X * 3 * GdC, _
                                                (Y1) - V1.Y * LdC + V2.Y * GdC + V3.Y *
3 * GdC, _
                                                (Z1) - V1.Z * LdC + V2.Z * GdC + V3.Z *
3 * GdC, Color1, Color1, 0, 0)



         m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 4) =
CreateLitVertex((X1), (Y1), (Z1), Color1, Color1, 0, 0)
         m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 5) =
CreateLitVertex((X1) - V1.X * LdC + V2.X * GdC + V3.X * 3 * GdC, _
```

```
                                                    (Y1) - V1.Y * LdC + V2.Y * GdC + V3.Y *
3 * GdC, _
                                                    (Z1) - V1.Z * LdC + V2.Z * GdC + V3.Z *
3 * GdC, Color2, Color2, 0, 0)

        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 6) =
CreateLitVertex((X1) - V1.X * LdC - V2.X * GdC + V3.X * 3 * GdC, _
                                                    (Y1) - V1.Y * LdC - V2.Y * GdC + V3.Y *
3 * GdC, _
                                                    (Z1) - V1.Z * LdC - V2.Z * GdC + V3.Z *
3 * GdC, Color2, Color2, 0, 0)


        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 7) =
CreateLitVertex((X1), (Y1), (Z1), Color1, Color1, 0, 0)
        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 8) =
CreateLitVertex((X1) - V1.X * LdC - V2.X * GdC + V3.X * 3 * GdC, _
                                                    (Y1) - V1.Y * LdC - V2.Y * GdC + V3.Y *
3 * GdC, _
                                                    (Z1) - V1.Z * LdC - V2.Z * GdC + V3.Z *
3 * GdC, Color1, Color1, 0, 0)

        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 9) =
CreateLitVertex((X1) - V1.X * LdC - V2.X * 3 * GdC + V3.X * GdC, _
                                                    (Y1) - V1.Y * LdC - V2.Y * 3 * GdC +
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC - V2.Z * 3 * GdC +
V3.Z * GdC, Color1, Color1, 0, 0)


        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 10) =
CreateLitVertex((X1), (Y1), (Z1), Color1, Color1, 0, 0)
        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 11) =
CreateLitVertex((X1) - V1.X * LdC - V2.X * 3 * GdC + V3.X * GdC, _
                                                    (Y1) - V1.Y * LdC - V2.Y * 3 * GdC +
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC - V2.Z * 3 * GdC +
V3.Z * GdC, Color2, Color2, 0, 0)

        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 12) =
CreateLitVertex((X1) - V1.X * LdC - V2.X * 3 * GdC - V3.X * GdC, _
                                                    (Y1) - V1.Y * LdC - V2.Y * 3 * GdC -
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC - V2.Z * 3 * GdC - V3.Z
* GdC, Color2, Color2, 0, 0)


        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 13) =
CreateLitVertex((X1), (Y1), (Z1), Color1, Color1, 0, 0)
        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 14) =
CreateLitVertex((X1) - V1.X * LdC - V2.X * 3 * GdC - V3.X * GdC, _
```

```
                                                    (Y1) - V1.Y * LdC - V2.Y * 3 * GdC -
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC - V2.Z * 3 * GdC - V3.Z
* GdC, Color1, Color1, 0, 0)

        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 15) =
CreateLitVertex((X1) - V1.X * LdC - V2.X * GdC - 3 * V3.X * GdC, _
                                                    (Y1) - V1.Y * LdC - V2.Y * GdC - 3 *
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC - V2.Z * GdC - 3 * V3.Z
* GdC, Color1, Color1, 0, 0)


        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 16) =
CreateLitVertex((X1), (Y1), (Z1), Color1, Color1, 0, 0)
        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 17) =
CreateLitVertex((X1) - V1.X * LdC - V2.X * GdC - 3 * V3.X * GdC, _
                                                    (Y1) - V1.Y * LdC - V2.Y * GdC - 3 *
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC - V2.Z * GdC - 3 * V3.Z
* GdC, Color2, Color2, 0, 0)

        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 18) =
CreateLitVertex((X1) - V1.X * LdC + V2.X * GdC - 3 * V3.X * GdC, _
                                                    (Y1) - V1.Y * LdC + V2.Y * GdC - 3 *
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC + V2.Z * GdC - 3 *
V3.Z * GdC, Color2, Color2, 0, 0)

        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 19) =
CreateLitVertex((X1), (Y1), (Z1), Color1, Color1, 0, 0)
        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 20) =
CreateLitVertex((X1) - V1.X * LdC + V2.X * 3 * GdC - V3.X * GdC, _
                                                    (Y1) - V1.Y * LdC + V2.Y * 3 * GdC -
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC + V2.Z * 3 * GdC -
V3.Z * GdC, Color1, Color1, 0, 0)

        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 21) =
CreateLitVertex((X1) - V1.X * LdC + V2.X * GdC - 3 * V3.X * GdC, _
                                                    (Y1) - V1.Y * LdC + V2.Y * GdC - 3 *
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC + V2.Z * GdC - 3 *
V3.Z * GdC, Color1, Color1, 0, 0)


        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 22) =
CreateLitVertex((X1), (Y1), (Z1), Color1, Color1, 0, 0)
        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 23) =
CreateLitVertex((X1) - V1.X * LdC + V2.X * 3 * GdC - V3.X * GdC, _
```

```
                                                    (Y1) - V1.Y * LdC + V2.Y * 3 * GdC -
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC + V2.Z * 3 * GdC -
V3.Z * GdC, Color2, Color2, 0, 0)

        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 24) =
CreateLitVertex((X1) - V1.X * LdC + V2.X * 3 * GdC + V3.X * GdC, _
                                                    (Y1) - V1.Y * LdC + V2.Y * 3 * GdC +
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC + V2.Z * 3 * GdC +
V3.Z * GdC, Color2, Color2, 0, 0)


        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 25) =
CreateLitVertex((X1) - V1.X * LdC, (Y1) - V1.Y * LdC, (Z1) - V1.Z * LdC, Color2, Color2, 0,
0)
        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 26) =
CreateLitVertex((X1) - V1.X * LdC + V2.X * 3 * GdC + V3.X * GdC, _
                                                    (Y1) - V1.Y * LdC + V2.Y * 3 * GdC +
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC + V2.Z * 3 * GdC +
V3.Z * GdC, Color2, Color2, 0, 0)

        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 27) =
CreateLitVertex((X1) - V1.X * LdC + V2.X * GdC + V3.X * 3 * GdC, _
                                                    (Y1) - V1.Y * LdC + V2.Y * GdC + V3.Y *
3 * GdC, _
                                                    (Z1) - V1.Z * LdC + V2.Z * GdC + V3.Z *
3 * GdC, Color2, Color2, 0, 0)


        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 28) =
CreateLitVertex((X1) - V1.X * LdC, (Y1) - V1.Y * LdC, (Z1) - V1.Z * LdC, Color1, Color1, 0,
0)
        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 29) =
CreateLitVertex((X1) - V1.X * LdC + V2.X * GdC + V3.X * 3 * GdC, _
                                                    (Y1) - V1.Y * LdC + V2.Y * GdC + V3.Y *
3 * GdC, _
                                                    (Z1) - V1.Z * LdC + V2.Z * GdC + V3.Z *
3 * GdC, Color1, Color1, 0, 0)

        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 30) =
CreateLitVertex((X1) - V1.X * LdC - V2.X * GdC + V3.X * 3 * GdC, _
                                                    (Y1) - V1.Y * LdC - V2.Y * GdC + V3.Y *
3 * GdC, _
                                                    (Z1) - V1.Z * LdC - V2.Z * GdC + V3.Z *
3 * GdC, Color1, Color1, 0, 0)
```

```
        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 31) =
CreateLitVertex((X1) - V1.X * LdC, (Y1) - V1.Y * LdC, (Z1) - V1.Z * LdC, Color2, Color2, 0,
0)
        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 32) =
CreateLitVertex((X1) - V1.X * LdC - V2.X * GdC + V3.X * 3 * GdC, _
                                                (Y1) - V1.Y * LdC - V2.Y * GdC + V3.Y *
3 * GdC, _
                                                (Z1) - V1.Z * LdC - V2.Z * GdC + V3.Z *
3 * GdC, Color2, Color2, 0, 0)

        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 33) =
CreateLitVertex((X1) - V1.X * LdC - V2.X * 3 * GdC + V3.X * GdC, _
                                                (Y1) - V1.Y * LdC - V2.Y * 3 * GdC +
V3.Y * GdC, _
                                                (Z1) - V1.Z * LdC - V2.Z * 3 * GdC +
V3.Z * GdC, Color2, Color2, 0, 0)


        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 34) =
CreateLitVertex((X1) - V1.X * LdC, (Y1) - V1.Y * LdC, (Z1) - V1.Z * LdC, Color1, Color1, 0,
0)
        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 35) =
CreateLitVertex((X1) - V1.X * LdC - V2.X * 3 * GdC + V3.X * GdC, _
                                                (Y1) - V1.Y * LdC - V2.Y * 3 * GdC +
V3.Y * GdC, _
                                                (Z1) - V1.Z * LdC - V2.Z * 3 * GdC +
V3.Z * GdC, Color1, Color1, 0, 0)

        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 36) =
CreateLitVertex((X1) - V1.X * LdC - V2.X * 3 * GdC - V3.X * GdC, _
                                                (Y1) - V1.Y * LdC - V2.Y * 3 * GdC -
V3.Y * GdC, _
                                                (Z1) - V1.Z * LdC - V2.Z * 3 * GdC - V3.Z
* GdC, Color1, Color1, 0, 0)


        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 37) =
CreateLitVertex((X1) - V1.X * LdC, (Y1) - V1.Y * LdC, (Z1) - V1.Z * LdC, Color2, Color2, 0,
0)
        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 38) =
CreateLitVertex((X1) - V1.X * LdC - V2.X * 3 * GdC - V3.X * GdC, _
                                                (Y1) - V1.Y * LdC - V2.Y * 3 * GdC -
V3.Y * GdC, _
                                                (Z1) - V1.Z * LdC - V2.Z * 3 * GdC - V3.Z
* GdC, Color2, Color2, 0, 0)

        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 39) =
CreateLitVertex((X1) - V1.X * LdC - V2.X * GdC - 3 * V3.X * GdC, _
                                                (Y1) - V1.Y * LdC - V2.Y * GdC - 3 *
V3.Y * GdC, _
```

```
                                                    (Z1) - V1.Z * LdC - V2.Z * GdC - 3 * V3.Z
* GdC, Color2, Color2, 0, 0)


        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 40) =
CreateLitVertex((X1) - V1.X * LdC, (Y1) - V1.Y * LdC, (Z1) - V1.Z * LdC, Color1, Color1, 0,
0)
        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 41) =
CreateLitVertex((X1) - V1.X * LdC - V2.X * GdC - 3 * V3.X * GdC, _
                                                    (Y1) - V1.Y * LdC - V2.Y * GdC - 3 *
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC - V2.Z * GdC - 3 * V3.Z
* GdC, Color1, Color1, 0, 0)

        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 42) =
CreateLitVertex((X1) - V1.X * LdC + V2.X * GdC - 3 * V3.X * GdC, _
                                                    (Y1) - V1.Y * LdC + V2.Y * GdC - 3 *
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC + V2.Z * GdC - 3 *
V3.Z * GdC, Color1, Color1, 0, 0)

        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 43) =
CreateLitVertex((X1) - V1.X * LdC, (Y1) - V1.Y * LdC, (Z1) - V1.Z * LdC, Color2, Color2, 0,
0)
        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 44) =
CreateLitVertex((X1) - V1.X * LdC + V2.X * 3 * GdC - V3.X * GdC, _
                                                    (Y1) - V1.Y * LdC + V2.Y * 3 * GdC -
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC + V2.Z * 3 * GdC -
V3.Z * GdC, Color2, Color2, 0, 0)

        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 45) =
CreateLitVertex((X1) - V1.X * LdC + V2.X * GdC - 3 * V3.X * GdC, _
                                                    (Y1) - V1.Y * LdC + V2.Y * GdC - 3 *
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC + V2.Z * GdC - 3 *
V3.Z * GdC, Color2, Color2, 0, 0)


        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 46) =
CreateLitVertex((X1) - V1.X * LdC, (Y1) - V1.Y * LdC, (Z1) - V1.Z * LdC, Color1, Color1, 0,
0)
        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 47) =
CreateLitVertex((X1) - V1.X * LdC + V2.X * 3 * GdC - V3.X * GdC, _
                                                    (Y1) - V1.Y * LdC + V2.Y * 3 * GdC -
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC + V2.Z * 3 * GdC -
V3.Z * GdC, Color1, Color1, 0, 0)

        m_VertexsDeCapsDeForcesBarra((ForçaEnCursAbsoluta - 1) * 48 + 48) =
CreateLitVertex((X1) - V1.X * LdC + V2.X * 3 * GdC + V3.X * GdC, _
```

```
                                                    (Y1) - V1.Y * LdC + V2.Y * 3 * GdC +
V3.Y * GdC, _
                                                    (Z1) - V1.Z * LdC + V2.Z * 3 * GdC +
V3.Z * GdC, Color1, Color1, 0, 0)


        End With
Etiqueta:
     Next ForçaEnCurs
   Next BarraEnCurs

Etiqueta2:


'Buffers Corresponents a les forces
'Set EstructuraVis.DadesDirectX.VBufferForcesBarraV1 =
EstructuraVis.DadesDirectX.D3DDeviceV1.CreateVertexBuffer _
(Len(m_VertexsDeForcesBarra(1)) * NumForces * 2, 0, TL_FVF, D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferForcesBarraV1, 0, _
Len(m_VertexsDeForcesBarra(1)) * NumForces * 2, 0, m_VertexsDeForcesBarra(1)

'Set EstructuraVis.DadesDirectX.VBufferForcesBarraV2 =
EstructuraVis.DadesDirectX.D3DDeviceV2.CreateVertexBuffer _
(Len(m_VertexsDeForcesBarra(1)) * NumForces * 2, 0, TL_FVF, D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferForcesBarraV2, 0, _
Len(m_VertexsDeForcesBarra(1)) * NumForces * 2, 0, m_VertexsDeForcesBarra(1)

Set EstructuraVis.DadesDirectX.VBufferForcesBarraV3 =
EstructuraVis.DadesDirectX.D3DDeviceV3.CreateVertexBuffer _
(Len(m_VertexsDeForcesBarra(1)) * NumForces * 2, 0, TL_FVF, D3DPOOL_DEFAULT)
D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferForcesBarraV3, 0, _
Len(m_VertexsDeForcesBarra(1)) * NumForces * 2, 0, m_VertexsDeForcesBarra(1)

'Set EstructuraVis.DadesDirectX.VBufferForcesBarraV4 =
EstructuraVis.DadesDirectX.D3DDeviceV4.CreateVertexBuffer _
(Len(m_VertexsDeForcesBarra(1)) * NumForces * 2, 0, TL_FVF, D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferForcesBarraV4, 0, _
Len(m_VertexsDeForcesBarra(1)) * NumForces * 2, 0, m_VertexsDeForcesBarra(1)

'Buffers Corresponents als caps de les forces

'Set EstructuraVis.DadesDirectX.VBufferCapsDeForcesBarraV1 =
EstructuraVis.DadesDirectX.D3DDeviceV1.CreateVertexBuffer _
(Len(m_VertexsDeCapsDeForcesBarra(1)) * NumForces * 48, 0, TL_FVF,
D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferCapsDeForcesBarraV1, 0,

_
Len(m_VertexsDeCapsDeForcesBarra(1)) * NumForces * 48, 0,
m_VertexsDeCapsDeForcesBarra(1)
```

```
'Set EstructuraVis.DadesDirectX.VBufferCapsDeForcesBarraV2 =
EstructuraVis.DadesDirectX.D3DDeviceV2.CreateVertexBuffer _
(Len(m_VertexsDeCapsDeForcesBarra(1)) * NumForces * 48, 0, TL_FVF,
D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferCapsDeForcesBarraV2, 0,
_
Len(m_VertexsDeCapsDeForcesBarra(1)) * NumForces * 48, 0,
m_VertexsDeCapsDeForcesBarra(1)

Set EstructuraVis.DadesDirectX.VBufferCapsDeForcesBarraV3 =
EstructuraVis.DadesDirectX.D3DDeviceV3.CreateVertexBuffer _
(Len(m_VertexsDeCapsDeForcesBarra(1)) * NumForces * 48, 0, TL_FVF,
D3DPOOL_DEFAULT)
D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferCapsDeForcesBarraV3, 0, _
Len(m_VertexsDeCapsDeForcesBarra(1)) * NumForces * 48, 0,
m_VertexsDeCapsDeForcesBarra(1)

'Set EstructuraVis.DadesDirectX.VBufferCapsDeForcesBarraV4 =
EstructuraVis.DadesDirectX.D3DDeviceV4.CreateVertexBuffer _
(Len(m_VertexsDeCapsDeForcesBarra(1)) * NumForces * 48, 0, TL_FVF,
D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferCapsDeForcesBarraV4, 0,
_
Len(m_VertexsDeCapsDeForcesBarra(1)) * NumForces * 48, 0,
m_VertexsDeCapsDeForcesBarra(1)


EstructuraVis.AuxFormularis.BufferForcesBarraEmplenat = True
End Sub



Sub AssignaGeometriaCarreguesRepartidesATotesLesVistes()

Dim BarraEnCurs As Long
Dim MaxBarra As Long
Dim SumaDeTotsElsComponents As Double
Dim ValorMaximDeCarrega As Double
Dim TamanyCarrega As Double
Dim Modul As Double
Dim VDirector As D3DVECTOR
Dim VNormal As D3DVECTOR
Dim VCostat As D3DVECTOR
Dim VAux1 As D3DVECTOR
Dim VAux2 As D3DVECTOR
Dim CarregaIni As Double, CarregaFi As Double
Dim CarregaVisible As Long
Dim CorrectorDeDistancia As Double
Dim Color1 As Long, Color2 As Long
```

```
Dim X1 As Double, X2 As Double, Y1 As Double, Y2 As Double, Z1 As Double, Z2 As
Double
Dim AccioEnCurs As Long
Dim Alfa As Double, Beta As Double, AlfaPrima As Double, AlfaDoblePrima As Double,
BetaPrima As Double
Dim NumTrams As Long, TramEnCurs As Long
Dim LZ1 As Double, LZ2 As Double, LX1 As Double, LX2 As Double, LY1 As Double, LY2
As Double
Dim Longitud As Double
Dim Contador As Long
Dim AmpleP As Double, LlargP As Double 'AMPLE I LLARG DEL PUNTER
Dim Delta As D3DVECTOR
Dim V1 As D3DVECTOR
Dim V2 As D3DVECTOR
Dim V3 As D3DVECTOR
Dim V4 As D3DVECTOR
Dim ModulPunter As Double
Dim YAux As Double
Dim ModAux As Double
Dim DeltaD As Double
Dim SHanInvertitElsNodes As Boolean

If EstructuraVis.ColBarra.Count = 0 Then Exit Sub

MaxBarra = EstructuraVis.ColBarra.Count
NumTrams = EstructuraVis.Preferencies.NumeroDeTramsDeLesCarreguesRepartides
AmpleP = EstructuraVis.Preferencies.AmplePunterCarregaRepartida
LlargP = EstructuraVis.Preferencies.LlargPunterCarregaRepartida


Contador = MaxBarra * (NumTrams + 2) * 2
ReDim m_VertexsDeCarregaRepartida(1 To Contador)
Contador = MaxBarra * (NumTrams + 1) * 3
ReDim m_VertexsDePuntersDeCarregaRepartida(1 To Contador)

If Principal.HiHaAlgunaCarregaRepartida = False Then
Set EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV1 = Nothing
Set EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV2 = Nothing
Set EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV3 = Nothing
Set EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV4 = Nothing
GoTo Etiqueta
End If

'Ressetegem els Búffers
Set EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV1 = Nothing
Set EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV2 = Nothing
Set EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV3 = Nothing
Set EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV4 = Nothing
```

```
  TamanyCarrega = EstructuraVis.Preferencies.CRTamany

  AccioEnCurs = EstructuraVis.AuxFormularis.AccioActivada

  'En primer lloc determino la càrrega màxima fent un recorregut per totes les barres, per a
la càrrega en curs

   ValorMaximDeCarrega = 0

   For BarraEnCurs = 1 To MaxBarra
      If EstructuraVis.ColBarra.Item(BarraEnCurs).ExisteixLaAccioRepartida(AccioEnCurs)
Then
         With
EstructuraVis.ColBarra.Item(BarraEnCurs).ColCarreguesRepartides.Item(Str(AccioEnCurs)
)
                  If ValorMaximDeCarrega < Abs(.CarregaIni) Then
                  ValorMaximDeCarrega = Abs(.CarregaIni)
                  End If

                  If ValorMaximDeCarrega < Abs(.CarregaFi) Then
                  ValorMaximDeCarrega = Abs(.CarregaFi)
                  End If

         End With
      End If
   Next BarraEnCurs


   CorrectorDeDistancia =
DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs
   CorrectorDeDistancia = 1



   For BarraEnCurs = 1 To MaxBarra

   'ASSIGNO ELS COLORS SEGONS LA CÀRREGA

   With EstructuraVis.ColAccio.Item(Str(AccioEnCurs))
   Color1 = D3DColorRGBA(.ColorRGBAccio(1), .ColorRGBAccio(2), .ColorRGBAccio(3),
255)
   Color2 = D3DColorRGBA(.ColorRGBAccio(1) / 2, .ColorRGBAccio(2) / 2,
.ColorRGBAccio(3) / 2, 255)


      If EstructuraVis.ColBarra.Item(BarraEnCurs).ExisteixLaAccioRepartida(AccioEnCurs)
Then
```

```
      If
EstructuraVis.ColBarra.Item(BarraEnCurs).ColCarreguesRepartides.Item(Str(AccioEnCurs)
).CarregaSeleccionada = True Then
      Color1 = D3DColorRGBA(.ColorRGBAccio(1) / 2, .ColorRGBAccio(2) / 2,
.ColorRGBAccio(3) / 2, 255)
      End If
      End If

   End With


   'FI ASSIGNACIÓ COLORS


      If EstructuraVis.ColBarra.Item(BarraEnCurs).ExisteixLaAccioRepartida(AccioEnCurs)
Then

         With
EstructuraVis.ColBarra.Item(BarraEnCurs).ColCarreguesRepartides.Item(Str(AccioEnCurs)
)
         CarregaIni = .CarregaIni / ValorMaximDeCarrega * _
         EstructuraVis.Preferencies.CRTamany * CorrectorDeDistancia
         CarregaFi = .CarregaFi / ValorMaximDeCarrega * _
         EstructuraVis.Preferencies.CRTamany * CorrectorDeDistancia
         End With

         With EstructuraVis.ColBarra.Item(BarraEnCurs)
         VDirector = TrobaVectorDirector(CreaVector(.X1, .Y1, .Z1), CreaVector(.X2, .Y2,
.Z2))
         D3DXVec3Normalize VDirector, VDirector

         VCostat = CreaVector(.VxOrto, .VyOrto, .VzOrto)
         D3DXVec3Normalize VCostat, VCostat

         D3DXVec3Cross VNormal, VDirector, VCostat
         D3DXVec3Normalize VNormal, VNormal

         X1 = .X1: X2 = .X2: Y1 = .Y1: Y2 = .Y2: Z1 = .Z1: Z2 = .Z2
         Longitud = .L
         AmpleP = EstructuraVis.Preferencies.AmplePunterCarregaRepartida
         LlargP = EstructuraVis.Preferencies.LlargPunterCarregaRepartida
         AmpleP = Longitud / NumTrams * AmpleP
         End With

            Select Case
EstructuraVis.ColBarra.Item((BarraEnCurs)).ColCarreguesRepartides.Item(Str(AccioEnCur
s)).TipusDeCarregaRepartida
            Case 2 'SEGONS ELS EIXOS LOCALS
```

```
            With
EstructuraVis.ColBarra.Item((BarraEnCurs)).ColCarreguesRepartides.Item(Str(AccioEnCur
s))

                For TramEnCurs = 0 To NumTrams

                LX1 = X1 + .ZCarregaIni * VDirector.X + (TramEnCurs / NumTrams) *
(Longitud - .ZCarregaFi - .ZCarregaIni) * VDirector.X
                LY1 = Y1 + .ZCarregaIni * VDirector.Y + (TramEnCurs / NumTrams) *
(Longitud - .ZCarregaFi - .ZCarregaIni) * VDirector.Y
                LZ1 = Z1 + .ZCarregaIni * VDirector.Z + (TramEnCurs / NumTrams) *
(Longitud - .ZCarregaFi - .ZCarregaIni) * VDirector.Z


                Alfa = .AngleAlfaCREixosLocals


                D3DXVec3Normalize VDirector, VDirector
                D3DXVec3Normalize VCostat, VCostat
                D3DXVec3Normalize VNormal, VNormal

                DeltaD = CarregaIni + (CarregaFi - CarregaIni) * (TramEnCurs / NumTrams)

                LX2 = LX1 + (VNormal.X * Sin(Alfa) + VCostat.X * Cos(Alfa)) * DeltaD
                LY2 = LY1 + (VNormal.Y * Sin(Alfa) + VCostat.Y * Cos(Alfa)) * DeltaD
                LZ2 = LZ1 + (VNormal.Z * Sin(Alfa) + VCostat.Z * Cos(Alfa)) * DeltaD

                Delta.X = VNormal.X * LlargP * Sin(Alfa) * Signe((DeltaD)) + VCostat.X *
LlargP * Cos(Alfa) * Signe((DeltaD))
                Delta.Y = VNormal.Y * LlargP * Sin(Alfa) * Signe((DeltaD)) + VCostat.Y *
LlargP * Cos(Alfa) * Signe((DeltaD))
                Delta.Z = VNormal.Z * LlargP * Sin(Alfa) * Signe((DeltaD)) + VCostat.Z * LlargP
* Cos(Alfa) * Signe((DeltaD))

                'LINIES
                m_VertexsDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams + 2) * 2) _
                + (TramEnCurs) * 2 + 1) = CreateLitVertex((LX1), (LY1), (LZ1), Color1, Color1,
0, 0)

                m_VertexsDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams + 2) * 2) _
                + (TramEnCurs) * 2 + 2) = CreateLitVertex((LX2), (LY2), (LZ2), Color1, Color1,
0, 0)

                'PUNTERS

                m_VertexsDePuntersDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams +
1) * 3) _
                + (TramEnCurs) * 3 + 1) = CreateLitVertex((LX1), (LY1), (LZ1), Color1, Color1,
0, 0)
```

```
            m_VertexsDePuntersDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams +
1) * 3) _
            + (TramEnCurs) * 3 + 2) = CreateLitVertex((LX1) + Delta.X + VDirector.X *
AmpleP, _
             (LY1) + Delta.Y + VDirector.Y * AmpleP, LZ1 + Delta.Z + VDirector.Z * AmpleP
_
            , Color1, Color1, 0, 0)

            m_VertexsDePuntersDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams +
1) * 3) _
            + (TramEnCurs) * 3 + 3) = CreateLitVertex((LX1) + Delta.X - VDirector.X *
AmpleP, _
             (LY1) + Delta.Y - VDirector.Y * AmpleP, LZ1 + Delta.Z - VDirector.Z * AmpleP
_
            , Color1, Color1, 0, 0)

            Next TramEnCurs


        LX1 = X1 + .ZCarregaIni * VDirector.X + _
        CarregaIni * Sin(.AngleAlfaCREixosLocals) * VNormal.X + _
        CarregaIni * Cos(.AngleAlfaCREixosLocals) * VCostat.X
        LY1 = Y1 + .ZCarregaIni * VDirector.Y + _
        CarregaIni * Sin(.AngleAlfaCREixosLocals) * VNormal.Y + _
        CarregaIni * Cos(.AngleAlfaCREixosLocals) * VCostat.Y
        LZ1 = Z1 + .ZCarregaIni * VDirector.Z + _
        CarregaIni * Sin(.AngleAlfaCREixosLocals) * VNormal.Z + _
        CarregaIni * Cos(.AngleAlfaCREixosLocals) * VCostat.Z


        LX2 = X2 - .ZCarregaFi * VDirector.X + _
        CarregaFi * Sin(.AngleAlfaCREixosLocals) * VNormal.X + _
        CarregaFi * Cos(.AngleAlfaCREixosLocals) * VCostat.X
        LY2 = Y2 - .ZCarregaFi * VDirector.Y + _
        CarregaFi * Sin(.AngleAlfaCREixosLocals) * VNormal.Y + _
        CarregaFi * Cos(.AngleAlfaCREixosLocals) * VCostat.Y
        LZ2 = Z2 - .ZCarregaFi * VDirector.Z + _
        CarregaFi * Sin(.AngleAlfaCREixosLocals) * VNormal.Z + _
        CarregaFi * Cos(.AngleAlfaCREixosLocals) * VCostat.Z

        m_VertexsDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams + 2) * 2) + _
        (NumTrams + 1) * 2 + 1) = CreateLitVertex((LX1), (LY1), (LZ1), Color1, Color1, 0,
0)

        m_VertexsDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams + 2) * 2) + _
        (NumTrams + 1) * 2 + 2) = CreateLitVertex((LX2), (LY2), (LZ2), Color1, Color1, 0,
0)
```

```
            End With

            Case 3 'SEGONS ELS EIXOS GLOBALS

            With
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).ColCarreguesRepartides.Item(Str(AccioEnC
urs))
            Alfa = .AngleAlfaCREixosGlobals
            Beta = .AngleBetaCREixosGlobals


               V2.X = X2 - X1: V2.Y = Y2 - Y1: V2.Z = Z2 - Z1
               D3DXVec3Normalize V2, V2

               V1.X = Cos(Alfa) * Cos(Beta)
               V1.Y = Sin(Alfa)
               V1.Z = Cos(Alfa) * Sin(Beta)
               D3DXVec3Normalize V1, V1

               D3DXVec3Cross V3, V2, V1
               D3DXVec3Normalize V3, V3

               D3DXVec3Cross V4, V1, V3
               D3DXVec3Normalize V4, V4

            For TramEnCurs = 0 To NumTrams

               Alfa = .AngleAlfaCREixosGlobals
               Beta = .AngleBetaCREixosGlobals

               LX1 = X1 + .ZCarregaIni * VDirector.X + (TramEnCurs / NumTrams) *
(Longitud - .ZCarregaFi - .ZCarregaIni) * VDirector.X
               LY1 = Y1 + .ZCarregaIni * VDirector.Y + (TramEnCurs / NumTrams) *
(Longitud - .ZCarregaFi - .ZCarregaIni) * VDirector.Y
               LZ1 = Z1 + .ZCarregaIni * VDirector.Z + (TramEnCurs / NumTrams) *
(Longitud - .ZCarregaFi - .ZCarregaIni) * VDirector.Z

               Delta.X = CarregaIni * Cos(Alfa) * Cos(Beta) + _
               (CarregaFi - CarregaIni) * Cos(Alfa) * Cos(Beta) * (TramEnCurs / NumTrams)

               Delta.Y = CarregaIni * Sin(Alfa) + _
               (CarregaFi - CarregaIni) * Sin(Alfa) * (TramEnCurs / NumTrams)

               Delta.Z = CarregaIni * Cos(Alfa) * Sin(Beta) + _
               (CarregaFi - CarregaIni) * Cos(Alfa) * Sin(Beta) * (TramEnCurs / NumTrams)


               LX2 = LX1 + Delta.X
               LY2 = LY1 + Delta.Y
               LZ2 = LZ1 + Delta.Z
```

```
            'LINIES
            m_VertexsDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams + 2) * 2) _
            + (TramEnCurs) * 2 + 1) = CreateLitVertex((LX1), (LY1), (LZ1), Color1, Color1,
0, 0)

            m_VertexsDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams + 2) * 2) _
            + (TramEnCurs) * 2 + 2) = CreateLitVertex((LX2), (LY2), (LZ2), Color1, Color1,
0, 0)

            'PUNTERS

            m_VertexsDePuntersDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams +
1) * 3) _
            + (TramEnCurs) * 3 + 1) = CreateLitVertex((LX1), (LY1), (LZ1), Color1, Color1,
0, 0)


            m_VertexsDePuntersDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams +
1) * 3) _
            + (TramEnCurs) * 3 + 2) = CreateLitVertex((LX1) + V1.X * LlargP *
Signe((Delta.X)) + V4.X * AmpleP, _
             (LY1) + V1.Y * LlargP * Signe((Delta.Y)) + V4.Y * AmpleP, LZ1 + V1.Z * LlargP
* Signe((Delta.Z)) + V4.Z * AmpleP _
            , Color1, Color1, 0, 0)

            m_VertexsDePuntersDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams +
1) * 3) _
            + (TramEnCurs) * 3 + 3) = CreateLitVertex((LX1) + V1.X * LlargP *
Signe((Delta.X)) - V4.X * AmpleP, _
             (LY1) + V1.Y * LlargP * Signe((Delta.Y)) - V4.Y * AmpleP, LZ1 + V1.Z * LlargP
* Signe((Delta.Z)) - V4.Z * AmpleP _
            , Color1, Color1, 0, 0)

            Next TramEnCurs

        Beta = .AngleBetaCREixosGlobals
        LX1 = X1 + .ZCarregaIni * VDirector.X + _
        CarregaIni * Cos(Alfa) * Cos(Beta)
        LY1 = Y1 + .ZCarregaIni * VDirector.Y + _
        CarregaIni * Sin(Alfa)
        LZ1 = Z1 + .ZCarregaIni * VDirector.Z + _
        CarregaIni * Cos(Alfa) * Sin(Beta)


        LX2 = X2 - .ZCarregaFi * VDirector.X + _
        CarregaFi * Cos(Alfa) * Cos(Beta)
        LY2 = Y2 - .ZCarregaFi * VDirector.Y + _
        CarregaFi * Sin(Alfa)
        LZ2 = Z2 - .ZCarregaFi * VDirector.Z + _
        CarregaFi * Cos(Alfa) * Sin(Beta)
```

```
            m_VertexsDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams + 2) * 2) + _
            (NumTrams + 1) * 2 + 1) = CreateLitVertex((LX1), (LY1), (LZ1), Color1, Color1, 0,
0)

            m_VertexsDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams + 2) * 2) + _
            (NumTrams + 1) * 2 + 2) = CreateLitVertex((LX2), (LY2), (LZ2), Color1, Color1, 0,
0)



            End With

            Case 4 'CARREGA PLANA

            With
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).ColCarreguesRepartides.Item(Str(AccioEnC
urs))
            Alfa = .AngleAlfaCREixosGlobals
            Beta = .AngleBetaCREixosGlobals


               V2.X = X2 - X1: V2.Y = Y2 - Y1: V2.Z = Z2 - Z1
               D3DXVec3Normalize V2, V2

               V1.X = Cos(Alfa) * Cos(Beta)
               V1.Y = Sin(Alfa)
               V1.Z = Cos(Alfa) * Sin(Beta)
               D3DXVec3Normalize V1, V1

               D3DXVec3Cross V3, V2, V1
               D3DXVec3Normalize V3, V3

               D3DXVec3Cross V4, V1, V3
               D3DXVec3Normalize V4, V4

               If Y1 < Y2 And .CarregaIni > 0 Then
               YAux = Y2
               Y2 = Y1
               Y1 = YAux
               SHanInvertitElsNodes = True
               End If

            For TramEnCurs = 0 To NumTrams

               Alfa = .AngleAlfaCREixosGlobals
               Beta = .AngleBetaCREixosGlobals

               LX1 = X1 + .ZCarregaIni * VDirector.X + (TramEnCurs / NumTrams) *
(Longitud - .ZCarregaFi - .ZCarregaIni) * VDirector.X
```

```
            LY1 = Y1 '+ .ZCarregaIni * VDirector.Y + (TramEnCurs / NumTrams) *
(Longitud - .ZCarregaFi - .ZCarregaIni) * VDirector.Y
            LZ1 = Z1 + .ZCarregaIni * VDirector.Z + (TramEnCurs / NumTrams) *
(Longitud - .ZCarregaFi - .ZCarregaIni) * VDirector.Z



            Delta.X = CarregaIni * Cos(Alfa) * Cos(Beta) + _
            (CarregaFi - CarregaIni) * Cos(Alfa) * Cos(Beta) * (TramEnCurs / NumTrams)

            Delta.Y = CarregaIni * Sin(Alfa) + _
            (CarregaFi - CarregaIni) * Sin(Alfa) * (TramEnCurs / NumTrams)

            Delta.Z = CarregaIni * Cos(Alfa) * Sin(Beta) + _
            (CarregaFi - CarregaIni) * Cos(Alfa) * Sin(Beta) * (TramEnCurs / NumTrams)


            LX2 = LX1 + Delta.X
            LY2 = LY1 + Delta.Y
            LZ2 = LZ1 + Delta.Z

            'LINIES
            m_VertexsDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams + 2) * 2) _
            + (TramEnCurs) * 2 + 1) = CreateLitVertex((LX1), (LY1), (LZ1), Color1, Color1,
0, 0)

            m_VertexsDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams + 2) * 2) _
            + (TramEnCurs) * 2 + 2) = CreateLitVertex((LX2), (LY2), (LZ2), Color1, Color1,
0, 0)

            'PUNTERS

            m_VertexsDePuntersDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams +
1) * 3) _
            + (TramEnCurs) * 3 + 1) = CreateLitVertex((LX1), (LY1), (LZ1), Color1, Color1,
0, 0)


            m_VertexsDePuntersDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams +
1) * 3) _
            + (TramEnCurs) * 3 + 2) = CreateLitVertex((LX1) + V1.X * LlargP *
Signe((Delta.X)) + V4.X * AmpleP, _
             (LY1) + V1.Y * LlargP * Signe((Delta.Y)) + V4.Y * AmpleP, LZ1 + V1.Z * LlargP
* Signe((Delta.Z)) + V4.Z * AmpleP _
            , Color1, Color1, 0, 0)

            m_VertexsDePuntersDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams +
1) * 3) _
            + (TramEnCurs) * 3 + 3) = CreateLitVertex((LX1) + V1.X * LlargP *
Signe((Delta.X)) - V4.X * AmpleP, _
```

```
            (LY1) + V1.Y * LlargP * Signe((Delta.Y)) - V4.Y * AmpleP, LZ1 + V1.Z * LlargP
* Signe((Delta.Z)) - V4.Z * AmpleP _
            , Color1, Color1, 0, 0)

            Next TramEnCurs

        Beta = .AngleBetaCREixosGlobals

        'If SHanInvertitElsNodes = False Then
        LX1 = X1 + .ZCarregaIni * VDirector.X + _
        CarregaIni * Cos(Alfa) * Cos(Beta)
        LY1 = Y1 + .ZCarregaIni * VDirector.Y + _
        CarregaIni * Sin(Alfa)
        LZ1 = Z1 + .ZCarregaIni * VDirector.Z + _
        CarregaIni * Cos(Alfa) * Sin(Beta)


        LX2 = X2 - .ZCarregaFi * VDirector.X + _
        CarregaFi * Cos(Alfa) * Cos(Beta)
        LY2 = Y1 + CarregaFi * Sin(Alfa)
        LZ2 = Z2 - .ZCarregaFi * VDirector.Z + _
        CarregaFi * Cos(Alfa) * Sin(Beta)


        'Else
        'LX1 = X2 + .ZCarregaIni * VDirector.X + _
        'CarregaIni * Cos(Alfa) * Cos(Beta)
        'LY1 = Y2 + .ZCarregaIni * VDirector.Y + _
        'CarregaIni * Sin(Alfa)
        'LZ1 = Z2 + .ZCarregaIni * VDirector.Z + _
        'CarregaIni * Cos(Alfa) * Sin(Beta)
        'LX2 = X1 - .ZCarregaFi * VDirector.X + _
        'CarregaFi * Cos(Alfa) * Cos(Beta)
        'LY2 = Y1 - .ZCarregaFi * VDirector.Y + _
        'CarregaFi * Sin(Alfa) * Signe((Delta.Y))
        'LZ2 = Z1 - .ZCarregaFi * VDirector.Z + _
        'CarregaFi * Cos(Alfa) * Sin(Beta)
        'End If


        m_VertexsDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams + 2) * 2) + _
        (NumTrams + 1) * 2 + 1) = CreateLitVertex((LX1), (LY1), (LZ1), Color1, Color1, 0,
0)

        m_VertexsDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams + 2) * 2) + _
        (NumTrams + 1) * 2 + 2) = CreateLitVertex((LX2), (LY2), (LZ2), Color1, Color1, 0,
0)



        End With
```

```
Case 5 'CÀRREGA AXIL

With
EstructuraVis.ColBarra.Item(Str(BarraEnCurs)).ColCarreguesRepartides.Item(Str(AccioEnC
urs))

V1.X = VNormal.X * CarregaIni + VDirector.X * .ZCarregaIni
V1.Y = VNormal.Y * CarregaIni + VDirector.Y * .ZCarregaIni
V1.Z = VNormal.Z * CarregaIni + VDirector.Z * .ZCarregaIni
'D3DXVec3Normalize V1, V1

V2.X = VNormal.X * CarregaFi + VDirector.X * (Longitud - .ZCarregaFi)
V2.Y = VNormal.Y * CarregaFi + VDirector.Y * (Longitud - .ZCarregaFi)
V2.Z = VNormal.Z * CarregaFi + VDirector.Z * (Longitud - .ZCarregaFi)
'D3DXVec3Normalize V2, V2

V3.X = V2.X - V1.X
V3.Y = V2.Y - V1.Y
V3.Z = V2.Z - V1.Z
D3DXVec3Normalize V3, V3

D3DXVec3Cross V4, V3, VCostat
D3DXVec3Normalize V4, V4


For TramEnCurs = 0 To NumTrams

LX1 = X1 + .ZCarregaIni * VDirector.X + (TramEnCurs / NumTrams) *
(Longitud - .ZCarregaFi - .ZCarregaIni) * VDirector.X
LY1 = Y1 + .ZCarregaIni * VDirector.Y + (TramEnCurs / NumTrams) *
(Longitud - .ZCarregaFi - .ZCarregaIni) * VDirector.Y
LZ1 = Z1 + .ZCarregaIni * VDirector.Z + (TramEnCurs / NumTrams) *
(Longitud - .ZCarregaFi - .ZCarregaIni) * VDirector.Z

    Delta.X = CarregaIni * VNormal.X + (CarregaFi - CarregaIni) * VNormal.X *
(TramEnCurs / NumTrams)
    Delta.Y = CarregaIni * VNormal.Y + (CarregaFi - CarregaIni) * VNormal.Y *
(TramEnCurs / NumTrams)
    Delta.Z = CarregaIni * VNormal.Z + (CarregaFi - CarregaIni) * VNormal.Z *
(TramEnCurs / NumTrams)

LX2 = LX1 + Delta.X
LY2 = LY1 + Delta.Y
LZ2 = LZ1 + Delta.Z

'LINIES
m_VertexsDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams + 2) * 2) _
+ (TramEnCurs) * 2 + 1) = CreateLitVertex((LX1), (LY1), (LZ1), Color1, Color1,
0, 0)
```

```
            m_VertexsDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams + 2) * 2) _
            + (TramEnCurs) * 2 + 2) = CreateLitVertex((LX2), (LY2), (LZ2), Color1, Color1,
0, 0)

            'PUNTERS

               'DEPENENT DE SI ÉS POSITIU O NEGATIU ES TINDRÀ
               If (CarregaIni + (CarregaFi - CarregaIni) * (TramEnCurs / NumTrams)) > 0
Then
               m_VertexsDePuntersDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams
+ 1) * 3) _
               + (TramEnCurs) * 3 + 1) = CreateLitVertex((LX2) + V3.X * LlargP, (LY2) _
               + V3.Y * LlargP, (LZ2) + V3.Z * LlargP, Color1, Color1, 0, 0)

               Else
               m_VertexsDePuntersDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams
+ 1) * 3) _
               + (TramEnCurs) * 3 + 1) = CreateLitVertex((LX2) - V3.X * LlargP, (LY2) _
               - V3.Y * LlargP, (LZ2) - V3.Z * LlargP, Color1, Color1, 0, 0)
               End If

               m_VertexsDePuntersDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams
+ 1) * 3) _
               + (TramEnCurs) * 3 + 2) = CreateLitVertex(LX2 + V4.X * AmpleP, _
               LY2 + V4.Y * AmpleP, LZ2 + V4.Z * AmpleP, _
               Color1, Color1, 0, 0)

               m_VertexsDePuntersDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams
+ 1) * 3) _
               + (TramEnCurs) * 3 + 3) = CreateLitVertex(LX2 - V4.X * AmpleP, _
               LY2 - V4.Y * AmpleP, LZ2 - V4.Z * AmpleP, _
               Color1, Color1, 0, 0)

               'PURGO UNA MICA PERQUE ES VEGI MILLOR
               If (TramEnCurs Mod 3) <> 0 Then

               m_VertexsDePuntersDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams
+ 1) * 3) _
               + (TramEnCurs) * 3 + 1) = CreateLitVertex(0, 0, 0, Color1, Color1, 0, 0)


               m_VertexsDePuntersDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams
+ 1) * 3) _
               + (TramEnCurs) * 3 + 2) = CreateLitVertex(0, 0, 0, _
               Color1, Color1, 0, 0)

               m_VertexsDePuntersDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams
+ 1) * 3) _
               + (TramEnCurs) * 3 + 3) = CreateLitVertex(0, 0, 0, _
               Color1, Color1, 0, 0)
```

```
            End If


            'FI PURGA

        Next TramEnCurs

    'LX1 = X1 + .ZCarregaIni * VDirector.X + _
    'CarregaIni * Sin(.AngleAlfaCREixosLocals) * VNormal.X + _
    'CarregaIni * Cos(.AngleAlfaCREixosLocals) * VCostat.X
    'LY1 = Y1 + .ZCarregaIni * VDirector.Y + _
    'CarregaIni * Sin(.AngleAlfaCREixosLocals) * VNormal.Y + _
    'CarregaIni * Cos(.AngleAlfaCREixosLocals) * VCostat.Y
    'LZ1 = Z1 + .ZCarregaIni * VDirector.Z + _
    'CarregaIni * Sin(.AngleAlfaCREixosLocals) * VNormal.Z + _
    'CarregaIni * Cos(.AngleAlfaCREixosLocals) * VCostat.Z


    'LX2 = X2 - .ZCarregaFi * VDirector.X + _
    'CarregaFi * Sin(.AngleAlfaCREixosLocals) * VNormal.X + _
    'CarregaFi * Cos(.AngleAlfaCREixosLocals) * VCostat.X
    'LY2 = Y2 - .ZCarregaFi * VDirector.Y + _
    'CarregaFi * Sin(.AngleAlfaCREixosLocals) * VNormal.Y + _
    'CarregaFi * Cos(.AngleAlfaCREixosLocals) * VCostat.Y
    'LZ2 = Z2 - .ZCarregaFi * VDirector.Z + _
    'CarregaFi * Sin(.AngleAlfaCREixosLocals) * VNormal.Z + _
    'CarregaFi * Cos(.AngleAlfaCREixosLocals) * VCostat.Z


    LX1 = X1 + .ZCarregaIni * VDirector.X + _
    CarregaIni * 1 * VNormal.X
    LY1 = Y1 + .ZCarregaIni * VDirector.Y + _
    CarregaIni * 1 * VNormal.Y
    LZ1 = Z1 + .ZCarregaIni * VDirector.Z + _
    CarregaIni * VNormal.Z


    LX2 = X2 - .ZCarregaFi * VDirector.X + _
    CarregaFi * 1 * VNormal.X
    LY2 = Y2 - .ZCarregaFi * VDirector.Y + _
    CarregaFi * 1 * VNormal.Y
    LZ2 = Z2 - .ZCarregaFi * VDirector.Z + _
    CarregaFi * 1 * VNormal.Z

    m_VertexsDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams + 2) * 2) + _
    (NumTrams + 1) * 2 + 1) = CreateLitVertex((LX1), (LY1), (LZ1), Color1, Color1, 0,
0)

    m_VertexsDeCarregaRepartida((BarraEnCurs - 1) * ((NumTrams + 2) * 2) + _
    (NumTrams + 1) * 2 + 2) = CreateLitVertex((LX2), (LY2), (LZ2), Color1, Color1, 0,
0)
```

```
            End With


          End Select
       End If

    Next BarraEnCurs

 'Ara omplim els buffers

Etiqueta:

EstructuraVis.AuxFormularis.DibuixaCarregaRepartida = True

'Set EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV1 =
EstructuraVis.DadesDirectX.D3DDeviceV1.CreateVertexBuffer _
(Len(m_VertexsDeCarregaRepartida(1)) * EstructuraVis.ColBarra.Count * 6, 0, TL_FVF,
D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV1, 0,
_
Len(m_VertexsDeCarregaRepartida(1)) * EstructuraVis.ColBarra.Count * 6, 0,
m_VertexsDeCarregaRepartida(1)

'Set EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV2 =
EstructuraVis.DadesDirectX.D3DDeviceV2.CreateVertexBuffer _
(Len(m_VertexsDeCarregaRepartida(1)) * EstructuraVis.ColBarra.Count * 6, 0, TL_FVF,
D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV2, 0,
_
Len(m_VertexsDeCarregaRepartida(1)) * EstructuraVis.ColBarra.Count * 6, 0,
m_VertexsDeCarregaRepartida(1)



'Set EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV3 =
EstructuraVis.DadesDirectX.D3DDeviceV3.CreateVertexBuffer _
(Len(m_VertexsDeCarregaRepartida(1)) * EstructuraVis.ColBarra.Count * 6, 0, TL_FVF,
D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV3, 0,
_
Len(m_VertexsDeCarregaRepartida(1)) * EstructuraVis.ColBarra.Count * 6, 0,
m_VertexsDeCarregaRepartida(1)

Contador = MaxBarra * (NumTrams + 2) * 2

Set EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV3 =
EstructuraVis.DadesDirectX.D3DDeviceV3.CreateVertexBuffer _
(Len(m_VertexsDeCarregaRepartida(1)) * Contador, 0, TL_FVF, D3DPOOL_DEFAULT)
```

```
D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV3, 0,
_
Len(m_VertexsDeCarregaRepartida(1)) * Contador, 0, m_VertexsDeCarregaRepartida(1)

Contador = MaxBarra * (NumTrams + 1) * 3
Set EstructuraVis.DadesDirectX.VBufferPunterCarreguesRepartidesV3 =
EstructuraVis.DadesDirectX.D3DDeviceV3.CreateVertexBuffer _
(Len(m_VertexsDePuntersDeCarregaRepartida(1)) * Contador, 0, TL_FVF,
D3DPOOL_DEFAULT)
D3DVertexBuffer8SetData
EstructuraVis.DadesDirectX.VBufferPunterCarreguesRepartidesV3, 0, _
Len(m_VertexsDePuntersDeCarregaRepartida(1)) * Contador, 0,
m_VertexsDePuntersDeCarregaRepartida(1)

'Set EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV4 =
EstructuraVis.DadesDirectX.D3DDeviceV4.CreateVertexBuffer _
(Len(m_VertexsDeCarregaRepartida(1)) * EstructuraVis.ColBarra.Count * 6, 0, TL_FVF,
D3DPOOL_DEFAULT)
'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferCarreguesRepartidesV4, 0,
_
Len(m_VertexsDeCarregaRepartida(1)) * EstructuraVis.ColBarra.Count * 6, 0,
m_VertexsDeCarregaRepartida(1)


EstructuraVis.AuxFormularis.BufferCarregaRepartidaEmplenat = True

End Sub



Sub AssignaGeometriaBarresaTotesLesVistes()
Dim BarraEnCurs As Long
Dim MaxBarra As Long
Dim Color1 As Long
'On Error GoTo Err:

MaxBarra = EstructuraVis.ColBarra.Count

        Set EstructuraVis.DadesDirectX.VBufferBarraV1 = Nothing
        Set EstructuraVis.DadesDirectX.VBufferBarraV2 = Nothing
        Set EstructuraVis.DadesDirectX.VBufferBarraV3 = Nothing
        Set EstructuraVis.DadesDirectX.VBufferBarraV4 = Nothing
        EstructuraVis.AuxFormularis.BufferBarresEmplenat = False


If MaxBarra = 0 Then Exit Sub

ReDim m_VertexsDeBarres(1 To EstructuraVis.ColBarra.Count * 2)

        For BarraEnCurs = 1 To MaxBarra
           If EstructuraVis.ColBarra.Item(BarraEnCurs).Seleccionat = True Then
```

```
            Color1 = EstructuraVis.Preferencies.ColorBarresSeleccionades
            Else
            Color1 = EstructuraVis.Preferencies.ColorBarres
            End If



            With EstructuraVis.ColBarra.Item(BarraEnCurs)
            m_VertexsDeBarres((BarraEnCurs - 1) * 2 + 1) = CreateLitVertex(.X1, .Y1, .Z1,
Color1, Color1, 0, 0)
            m_VertexsDeBarres((BarraEnCurs - 1) * 2 + 2) = CreateLitVertex(.X2, .Y2, .Z2,
Color1, Color1, 0, 0)
            End With
        Next BarraEnCurs



        Set EstructuraVis.DadesDirectX.VBufferBarraV3 =
EstructuraVis.DadesDirectX.D3DDeviceV3.CreateVertexBuffer(Len(m_VertexsDeBarres(1))
* MaxBarra * 2, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferBarraV3, 0,
Len(m_VertexsDeBarres(1)) * MaxBarra * 2, 0, m_VertexsDeBarres(1)


        EstructuraVis.AuxFormularis.BufferBarresEmplenat = True
End Sub
Sub AssignaGeometriaAlFormulariPissarra()


Dim BarraEnCurs As Long
Dim MaxBarra As Long
Dim VDirector(1 To 3) As Double, VD As D3DVECTOR
Dim VOrto(1 To 3) As Double, VC As D3DVECTOR
Dim VAux(1 To 3) As Double, VN As D3DVECTOR
Dim Modul As Double
Dim VectorAuxiliar(100) As Double
Dim P(1 To 12) As D3DVECTOR
Dim PC(1 To 40) As D3DVECTOR
If EstructuraVis.ColBarra.Count = 0 Then Exit Sub
Dim i As Long
Dim Color As Long, Ample As Single


Set EstructuraVis.DadesDirectX.vbufferpissarra = Nothing
MaxBarra = EstructuraVis.ColBarra.Count

ReDim m_VertexsDePissarra(1 To 36 * MaxBarra)
ReDim m_VertexsDePissarra2(1 To 18 * MaxBarra)
ReDim m_VertexsDePissarra3(1 To 240 * MaxBarra)
```

```
ReDim m_VertexsDePissarraS(1 To 36 * MaxBarra)
ReDim m_VertexsDePissarraS2(1 To 18 * MaxBarra)
ReDim m_VertexsDePissarraS3(1 To 240 * MaxBarra)

ReDim m_VertexsDePissarraA(1 To 36 * MaxBarra)
ReDim m_VertexsDePissarraA2(1 To 18 * MaxBarra)
ReDim m_VertexsDePissarraA3(1 To 240 * MaxBarra)

ReDim m_VertexsDePissarra4(1 To 18 * MaxBarra)
ReDim m_VertexsDePissarraS4(1 To 18 * MaxBarra)
ReDim m_VertexsDePissarraA4(1 To 18 * MaxBarra)

ReDim m_VertexsDePissarra5(1 To 12 * MaxBarra)
ReDim m_VertexsDePissarraS5(1 To 12 * MaxBarra)
ReDim m_VertexsDePissarraA5(1 To 12 * MaxBarra)



For BarraEnCurs = 1 To MaxBarra
Select Case EstructuraVis.ColBarra.Item(BarraEnCurs).TipusPerfil

   Case 1 'PERFIL RECTANGULAR

      Select Case EstructuraVis.ColBarra.Item(BarraEnCurs).Material

         Case "Formigó"
            With EstructuraVis.ColBarra.Item(BarraEnCurs)

            VDirector(1) = .X2 - .X1: VDirector(2) = .Y2 - .Y1: VDirector(3) = .Z2 - .Z1
            Modul = (VDirector(1) ^ 2 + VDirector(2) ^ 2 + VDirector(3) ^ 2) ^ 0.5
            VDirector(1) = VDirector(1) / Modul: VDirector(2) = VDirector(2) / Modul:
VDirector(3) = VDirector(3) / Modul

            VOrto(1) = .VxOrto: VOrto(2) = .VyOrto: VOrto(3) = .VzOrto
            Modul = (VOrto(1) ^ 2 + VOrto(2) ^ 2 + VOrto(3) ^ 2) ^ 0.5
            VOrto(1) = VOrto(1) / Modul: VOrto(2) = VOrto(2) / Modul: VOrto(3) = VOrto(3) /
Modul

            'Faig el producte vectorial
            VAux(1) = VDirector(2) * VOrto(3) - VDirector(3) * VOrto(2)
            VAux(2) = VDirector(3) * VOrto(1) - VDirector(1) * VOrto(3)
            VAux(3) = VDirector(1) * VOrto(2) - VDirector(2) * VOrto(1)

            Modul = (VAux(1) ^ 2 + VAux(2) ^ 2 + VAux(3) ^ 2) ^ 0.5
            VAux(1) = VAux(1) / Modul: VAux(2) = VAux(2) / Modul: VAux(3) = VAux(3) /
Modul

            'EnFront
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 1) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
```

```
            .Y1 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), QBColor(15), 0, 0)
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 3) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), QBColor(15), .L * 2, 0)
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 2) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), QBColor(15), 0, .h * 2)


            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 4) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, 0)
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 6) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, .h * 2)
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 5) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), QBColor(15), 0, .h * 2)


            'Darrere
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 7) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, QBColor(15), RGB(255, 255, 255), 0, 0)
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 8) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, QBColor(15), RGB(255, 255, 255), .L * 2, 0)
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 9) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), QBColor(15), 0, .h * 2)


            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 10) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, QBColor(15), RGB(255, 255, 255), .L * 2, 0)
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 11) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y2 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, .h * 2)
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 12) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
```

```
        .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), QBColor(15), 0, .h * 2)

        'Costat Dret
        m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 13) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
RGB(220, 220, 220), RGB(220, 220, 220), .B * 2, 0)
        m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 14) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y2 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, RGB(220, 220, 220), RGB(220, 220, 220), 0, 0)
        m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 15) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
        .Y2 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(220, 220, 220), RGB(220, 220, 220), 0, .h * 2)

        m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 16) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
RGB(220, 220, 220), RGB(220, 220, 220), .B * 2, 0)
        m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 17) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
        .Y2 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(220, 220, 220), RGB(220, 220, 220), 0, .h * 2)
        m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 18) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
        .Y2 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(220, 220, 220), RGB(220, 220, 220), .B * 2, .h * 2)

        'Costat Esquerre
        m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 19) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y1 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
RGB(220, 220, 220), RGB(220, 220, 220), .B * 2, 0)
        m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 20) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y1 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, RGB(220, 220, 220), RGB(220, 220, 220), 0, 0)
        m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 21) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
        .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(220, 220, 220), RGB(220, 220, 220), 0, .h * 2)

        m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 22) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y1 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
RGB(220, 220, 220), RGB(220, 220, 220), .B * 2, 0)
        m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 23) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
```

```
            .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(220, 220, 220), RGB(220, 220, 220), 0, .h * 2)
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 24) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(220, 220, 220), RGB(220, 220, 220), .B * 2, .h * 2)

            'Dalt
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 25) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y1 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
RGB(200, 200, 200), RGB(200, 200, 200), 0, 0)
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 26) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
RGB(200, 200, 200), RGB(200, 200, 200), .L * 2, 0)
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 27) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, RGB(200, 200, 200), RGB(200, 200, 200), 0, .B * 2)

            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 29) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, RGB(200, 200, 200), RGB(200, 200, 200), .L * 2, .B * 2)
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 28) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
RGB(200, 200, 200), RGB(200, 200, 200), .L * 2, 0)
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 30) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, RGB(200, 200, 200), RGB(200, 200, 200), 0, .B * 2)

            'Baix
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 31) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(200, 200, 200), RGB(200, 200, 200), 0, 0)
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 33) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(200, 200, 200), RGB(200, 200, 200), .L * 2, 0)
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 32) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(200, 200, 200), RGB(200, 200, 200), 0, .h * 2)

            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 36) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
```

```
            .Y2 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(200, 200, 200), RGB(200, 200, 200), .L * 2, .h * 2)
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 34) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(200, 200, 200), RGB(200, 200, 200), .L * 2, 0)
            m_VertexsDePissarra((BarraEnCurs - 1) * 36 + 35) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(200, 200, 200), RGB(200, 200, 200), 0, .h * 2)


        End With
        Case "Acer"
        With EstructuraVis.ColBarra.Item(BarraEnCurs)

            VDirector(1) = .X2 - .X1: VDirector(2) = .Y2 - .Y1: VDirector(3) = .Z2 - .Z1
            Modul = (VDirector(1) ^ 2 + VDirector(2) ^ 2 + VDirector(3) ^ 2) ^ 0.5
            VDirector(1) = VDirector(1) / Modul: VDirector(2) = VDirector(2) / Modul:
VDirector(3) = VDirector(3) / Modul

            VOrto(1) = .VxOrto: VOrto(2) = .VyOrto: VOrto(3) = .VzOrto
            Modul = (VOrto(1) ^ 2 + VOrto(2) ^ 2 + VOrto(3) ^ 2) ^ 0.5
            VOrto(1) = VOrto(1) / Modul: VOrto(2) = VOrto(2) / Modul: VOrto(3) = VOrto(3) /
Modul

            'Faig el producte vectorial
            VAux(1) = VDirector(2) * VOrto(3) - VDirector(3) * VOrto(2)
            VAux(2) = VDirector(3) * VOrto(1) - VDirector(1) * VOrto(3)
            VAux(3) = VDirector(1) * VOrto(2) - VDirector(2) * VOrto(1)

            Modul = (VAux(1) ^ 2 + VAux(2) ^ 2 + VAux(3) ^ 2) ^ 0.5
            VAux(1) = VAux(1) / Modul: VAux(2) = VAux(2) / Modul: VAux(3) = VAux(3) /
Modul

            'EnFront
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 1) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y1 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), QBColor(15), 0, 0)
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 3) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), QBColor(15), .L * 2, 0)
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 2) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), 0, .h * 2)
```

```
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 4) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, 0)
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 6) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, .h * 2)
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 5) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), 0, .h * 2)


            'Darrere
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 7) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, QBColor(15), RGB(255, 255, 255), 0, 0)
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 8) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, QBColor(15), RGB(255, 255, 255), .L * 2, 0)
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 9) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), 0, .h * 2)


            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 10) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, QBColor(15), RGB(255, 255, 255), .L * 2, 0)
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 11) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y2 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, .h * 2)
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 12) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), 0, .h * 2)

            'Costat Dret
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 13) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), 0, 0)
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 14) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, QBColor(15), RGB(255, 255, 255), 0, .h * 2)
```

```
        m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 15) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
        .Y2 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), .L * 2, 0)

        m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 17) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, .h * 2)
        m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 18) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
        .Y2 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), .L * 2, 0)
        m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 16) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
        .Y2 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), 0, .h * 2)

        'Costat Esquerre
        m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 20) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y1 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), 0, 0)
        m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 21) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y1 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, QBColor(15), RGB(255, 255, 255), 0, .h * 2)
        m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 19) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
        .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, 0)

        m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 22) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y1 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, .h * 2)
        m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 23) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
        .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, 0)
        m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 24) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
        .Y1 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), 0, .h * 2)

        'Dalt
        m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 25) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y1 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), 0, .h * 2)
```

```
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 26) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, 0)
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 27) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)

            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 29) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, QBColor(15), RGB(255, 255, 255), .L * 2, .h * 2)
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 28) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, 0)
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 30) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, RGB(150, 150, 150), RGB(150, 150, 150), 0, .h * 2)

            'Baix
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 31) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), 0, 0)
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 33) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, 0)
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 32) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), 0, .h * 2)

            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 36) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y2 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, .h * 2)
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 34) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, 0)
            m_VertexsDePissarraS((BarraEnCurs - 1) * 36 + 35) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), 0, .h * 2)


        End With
```

```
        Case Else
            With EstructuraVis.ColBarra.Item(BarraEnCurs)

            VDirector(1) = .X2 - .X1: VDirector(2) = .Y2 - .Y1: VDirector(3) = .Z2 - .Z1
            Modul = (VDirector(1) ^ 2 + VDirector(2) ^ 2 + VDirector(3) ^ 2) ^ 0.5
            VDirector(1) = VDirector(1) / Modul: VDirector(2) = VDirector(2) / Modul:
VDirector(3) = VDirector(3) / Modul

            VOrto(1) = .VxOrto: VOrto(2) = .VyOrto: VOrto(3) = .VzOrto
            Modul = (VOrto(1) ^ 2 + VOrto(2) ^ 2 + VOrto(3) ^ 2) ^ 0.5
            VOrto(1) = VOrto(1) / Modul: VOrto(2) = VOrto(2) / Modul: VOrto(3) = VOrto(3) /
Modul

            'Faig el producte vectorial
            VAux(1) = VDirector(2) * VOrto(3) - VDirector(3) * VOrto(2)
            VAux(2) = VDirector(3) * VOrto(1) - VDirector(1) * VOrto(3)
            VAux(3) = VDirector(1) * VOrto(2) - VDirector(2) * VOrto(1)

            Modul = (VAux(1) ^ 2 + VAux(2) ^ 2 + VAux(3) ^ 2) ^ 0.5
            VAux(1) = VAux(1) / Modul: VAux(2) = VAux(2) / Modul: VAux(3) = VAux(3) /
Modul

            'EnFront
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 1) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y1 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), QBColor(15), 0, 0)
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 3) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), QBColor(15), .L * 2, 0)
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 2) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), 0, .h * 2)


            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 4) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, 0)
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 6) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, .h * 2)
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 5) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), 0, .h * 2)
```

```
            'Darrere
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 7) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, QBColor(15), RGB(255, 255, 255), 0, 0)
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 8) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, QBColor(15), RGB(255, 255, 255), .L * 2, 0)
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 9) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), 0, .h * 2)


            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 10) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, QBColor(15), RGB(255, 255, 255), .L * 2, 0)
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 11) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y2 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, .h * 2)
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 12) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), 0, .h * 2)

            'Costat Dret
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 13) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), 0, 0)
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 14) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, QBColor(15), RGB(255, 255, 255), 0, .h * 2)
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 15) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y2 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), .L * 2, 0)

            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 17) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, .h * 2)
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 18) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y2 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), .L * 2, 0)
```

```
        m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 16) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
        .Y2 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), 0, .h * 2)

        'Costat Esquerre
        m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 20) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y1 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), 0, 0)
        m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 21) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y1 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, QBColor(15), RGB(255, 255, 255), 0, .h * 2)
        m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 19) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
        .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, 0)

        m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 22) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y1 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, .h * 2)
        m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 23) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
        .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, 0)
        m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 24) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
        .Y1 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), 0, .h * 2)

        'Dalt
        m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 25) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y1 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), 0, .h * 2)
        m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 26) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, 0)
        m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 27) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y1 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)

        m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 29) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
        .Y2 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, QBColor(15), RGB(255, 255, 255), .L * 2, .h * 2)
```

```
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 28) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 + VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, 0)
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 30) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 + VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 + VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 + VAux(3) * .h /
2, RGB(150, 150, 150), RGB(150, 150, 150), 0, .h * 2)

            'Baix
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 31) = CreateLitVertex(.X1 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), 0, 0)
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 33) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, 0)
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 32) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), 0, .h * 2)

            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 36) = CreateLitVertex(.X2 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y2 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, .h * 2)
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 34) = CreateLitVertex(.X2 -
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y2 - VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z2 - VOrto(3) * .B / 2 - VAux(3) * .h / 2,
QBColor(15), RGB(255, 255, 255), .L * 2, 0)
            m_VertexsDePissarraA((BarraEnCurs - 1) * 36 + 35) = CreateLitVertex(.X1 +
VOrto(1) * .B / 2 - VAux(1) * .h / 2, _
            .Y1 + VOrto(2) * .B / 2 - VAux(2) * .h / 2, .Z1 + VOrto(3) * .B / 2 - VAux(3) * .h / 2,
RGB(150, 150, 150), RGB(150, 150, 150), 0, .h * 2)


          End With
        End Select

    Case 2 'PERFIL CIRCULAR
      With EstructuraVis.ColBarra.Item(BarraEnCurs)
       VD = CreaVector(.X2 - .X1, .Y2 - .Y1, .Z2 - .Z1)
       D3DXVec3Normalize VD, VD

       VC = CreaVector(.VxOrto, .VyOrto, .VzOrto)
       D3DXVec3Normalize VC, VC

       D3DXVec3Cross VN, VD, VC
       D3DXVec3Normalize VN, VN
```

```
For i = 1 To 20
PC(i).X = .X1 + .R * Cos(Pi / 10 * (i - 1)) * VC.X + .R * Sin(Pi / 10 * (i - 1)) * VN.X
PC(i).Y = .Y1 + .R * Cos(Pi / 10 * (i - 1)) * VC.Y + .R * Sin(Pi / 10 * (i - 1)) * VN.Y
PC(i).Z = .Z1 + .R * Cos(Pi / 10 * (i - 1)) * VC.Z + .R * Sin(Pi / 10 * (i - 1)) * VN.Z
Next i

For i = 1 To 20
PC(i + 20).X = .X2 + .R * Cos(Pi / 10 * (i - 1)) * VC.X + .R * Sin(Pi / 10 * (i - 1)) * VN.X
PC(i + 20).Y = .Y2 + .R * Cos(Pi / 10 * (i - 1)) * VC.Y + .R * Sin(Pi / 10 * (i - 1)) * VN.Y
PC(i + 20).Z = .Z2 + .R * Cos(Pi / 10 * (i - 1)) * VC.Z + .R * Sin(Pi / 10 * (i - 1)) * VN.Z
Next i


Select Case EstructuraVis.ColBarra.Item(BarraEnCurs).Material

   Case "Formigó"

   'LATERAL
   For i = 1 To 20
   Color = D3DColorRGBA(150 + Abs(100 * Cos(Pi / 10 * i)), 150 + Abs(100 * Cos(Pi /
10 * i)), 150 + Abs(100 * Cos(Pi / 10 * i)), 255) 'DE MOMENT
   Ample = 1 / 20 * 4 * Pi * .R
     If i < 20 Then

      m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + (i - 1) * 6 + 1) =
CreateLitVertex(PC(i).X, PC(i).Y, PC(i).Z, Color, Color, 0, Ample * (i - 1))
      m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + (i - 1) * 6 + 2) =
CreateLitVertex(PC(i + 1).X, PC(i + 1).Y, PC(i + 1).Z, Color, Color, 0, Ample * i)
      m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + (i - 1) * 6 + 3) =
CreateLitVertex(PC(i + 20).X, PC(i + 20).Y, PC(i + 20).Z, Color, Color, .L * 2, Ample * (i - 1))

      m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + (i - 1) * 6 + 4) =
CreateLitVertex(PC(i + 1).X, PC(i + 1).Y, PC(i + 1).Z, Color, Color, 0, Ample * (i))
      m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + (i - 1) * 6 + 5) =
CreateLitVertex(PC(i + 20).X, PC(i + 20).Y, PC(i + 20).Z, Color, Color, .L * 2, Ample * (i - 1))
      m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + (i - 1) * 6 + 6) =
CreateLitVertex(PC(i + 21).X, PC(i + 21).Y, PC(i + 21).Z, Color, Color, .L * 2, Ample * (i))

      ElseIf i = 20 Then

      m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + (i - 1) * 6 + 1) =
CreateLitVertex(PC(20).X, PC(20).Y, PC(20).Z, Color, Color, 0, 0)
      m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + (i - 1) * 6 + 2) =
CreateLitVertex(PC(1).X, PC(1).Y, PC(1).Z, Color, Color, Ample, 0)
      m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + (i - 1) * 6 + 3) =
CreateLitVertex(PC(40).X, PC(40).Y, PC(40).Z, Color, Color, 0, .L * 2)
```

```
            m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + (i - 1) * 6 + 4) =
CreateLitVertex(PC(1).X, PC(1).Y, PC(1).Z, Color, Color, Ample, 0)
            m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + (i - 1) * 6 + 5) =
CreateLitVertex(PC(40).X, PC(40).Y, PC(40).Z, Color, Color, 0, .L * 2)
            m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + (i - 1) * 6 + 6) =
CreateLitVertex(PC(21).X, PC(21).Y, PC(21).Z, Color, Color, Ample, .L * 2)

          End If
        Next i


        'TAPA SUPERIOR

        For i = 1 To 20
        Color = D3DColorRGBA(200, 200, 200, 255)

          If i < 20 Then

            m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + 120 + (i - 1) * 3 + 1) =
CreateLitVertex(.X1, .Y1, .Z1, Color, Color, 0, 0)
            m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + 120 + (i - 1) * 3 + 2) =
CreateLitVertex(PC(i).X, PC(i).Y, PC(i).Z, Color, Color, Abs(Sin(Pi / 10 * i) * 2 * .R),
Abs(Cos(Pi / 10 * i) * 2 * .R))
            m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + 120 + (i - 1) * 3 + 3) =
CreateLitVertex(PC(i + 1).X, PC(i + 1).Y, PC(i + 1).Z, Color, Color, Abs(Sin(Pi / 10 * (i + 1)))
* 2 * .R, Abs(Cos(Pi / 10 * (i + 1)) * 2 * .R))


          ElseIf i = 20 Then

            m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + 120 + (i - 1) * 3 + 1) =
CreateLitVertex(.X1, .Y1, .Z1, Color, Color, 0, 0)
            m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + 120 + (i - 1) * 3 + 2) =
CreateLitVertex(PC(i).X, PC(i).Y, PC(i).Z, Color, Color, Abs(Sin(Pi / 10 * i) * 2 * .R),
Abs(Cos(Pi / 10 * i) * 2 * .R))
            m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + 120 + (i - 1) * 3 + 3) =
CreateLitVertex(PC(1).X, PC(1).Y, PC(1).Z, Color, Color, Abs(Sin(Pi / 10 * (i + 1))) * 2 * .R,
Abs(Cos(Pi / 10 * (i + 1)) * 2 * .R))


          End If
        Next i


         'TAPA INFERIOR

        For i = 1 To 20
        Color = D3DColorRGBA(200, 200, 200, 255)

          If i < 20 Then
```

```
            m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + 180 + (i - 1) * 3 + 1) =
CreateLitVertex(.X2, .Y2, .Z2, Color, Color, 0, 0)
            m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + 180 + (i - 1) * 3 + 2) =
CreateLitVertex(PC(i + 20).X, PC(i + 20).Y, PC(i + 20).Z, Color, Color, Abs(Sin(Pi / 10 * i) *
2 * .R), Abs(Cos(Pi / 10 * i) * 2 * .R))
            m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + 180 + (i - 1) * 3 + 3) =
CreateLitVertex(PC(i + 21).X, PC(i + 21).Y, PC(i + 21).Z, Color, Color, Abs(Sin(Pi / 10 * (i +
1))) * 2 * .R, Abs(Cos(Pi / 10 * (i + 1)) * 2 * .R))


            ElseIf i = 20 Then

            m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + 180 + (i - 1) * 3 + 1) =
CreateLitVertex(.X2, .Y2, .Z2, Color, Color, 0, 0)
            m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + 180 + (i - 1) * 3 + 2) =
CreateLitVertex(PC(i + 20).X, PC(i + 20).Y, PC(i + 20).Z, Color, Color, Abs(Sin(Pi / 10 * i) *
2 * .R), Abs(Cos(Pi / 10 * i) * 2 * .R))
            m_VertexsDePissarra3((BarraEnCurs - 1) * 240 + 180 + (i - 1) * 3 + 3) =
CreateLitVertex(PC(21).X, PC(21).Y, PC(21).Z, Color, Color, Abs(Sin(Pi / 10 * (i + 1))) * 2 *
.R, Abs(Cos(Pi / 10 * (i + 1)) * 2 * .R))



            End If
        Next i


        Case "Acer"



        Case Else

        End Select



    End With



Case 3
    With EstructuraVis.ColBarra.Item(BarraEnCurs)
```

```
VD = CreaVector(.X2 - .X1, .Y2 - .Y1, .Z2 - .Z1)
D3DXVec3Normalize VD, VD

VC = CreaVector(.VxOrto, .VyOrto, .VzOrto)
D3DXVec3Normalize VC, VC

D3DXVec3Cross VN, VD, VC
D3DXVec3Normalize VN, VN

P(1).X = .X1 + .h / 2 * VN.X - .B / 2 * VC.X
P(1).Y = .Y1 + .h / 2 * VN.Y - .B / 2 * VC.Y
P(1).Z = .Z1 + .h / 2 * VN.Z - .B / 2 * VC.Z

P(2).X = .X1 + .h / 2 * VN.X
P(2).Y = .Y1 + .h / 2 * VN.Y
P(2).Z = .Z1 + .h / 2 * VN.Z

P(3).X = .X1 + .h / 2 * VN.X + .B / 2 * VC.X
P(3).Y = .Y1 + .h / 2 * VN.Y + .B / 2 * VC.Y
P(3).Z = .Z1 + .h / 2 * VN.Z + .B / 2 * VC.Z

P(4).X = .X1 - .h / 2 * VN.X - .B / 2 * VC.X
P(4).Y = .Y1 - .h / 2 * VN.Y - .B / 2 * VC.Y
P(4).Z = .Z1 - .h / 2 * VN.Z - .B / 2 * VC.Z

P(5).X = .X1 - .h / 2 * VN.X
P(5).Y = .Y1 - .h / 2 * VN.Y
P(5).Z = .Z1 - .h / 2 * VN.Z

P(6).X = .X1 - .h / 2 * VN.X + .B / 2 * VC.X
P(6).Y = .Y1 - .h / 2 * VN.Y + .B / 2 * VC.Y
P(6).Z = .Z1 - .h / 2 * VN.Z + .B / 2 * VC.Z

P(7).X = .X2 + .h / 2 * VN.X - .B / 2 * VC.X
P(7).Y = .Y2 + .h / 2 * VN.Y - .B / 2 * VC.Y
P(7).Z = .Z2 + .h / 2 * VN.Z - .B / 2 * VC.Z

P(8).X = .X2 + .h / 2 * VN.X
P(8).Y = .Y2 + .h / 2 * VN.Y
P(8).Z = .Z2 + .h / 2 * VN.Z

P(9).X = .X2 + .h / 2 * VN.X + .B / 2 * VC.X
P(9).Y = .Y2 + .h / 2 * VN.Y + .B / 2 * VC.Y
P(9).Z = .Z2 + .h / 2 * VN.Z + .B / 2 * VC.Z

P(10).X = .X2 - .h / 2 * VN.X - .B / 2 * VC.X
P(10).Y = .Y2 - .h / 2 * VN.Y - .B / 2 * VC.Y
P(10).Z = .Z2 - .h / 2 * VN.Z - .B / 2 * VC.Z

P(11).X = .X2 - .h / 2 * VN.X
P(11).Y = .Y2 - .h / 2 * VN.Y
```

```
P(11).Z = .Z2 - .h / 2 * VN.Z

P(12).X = .X2 - .h / 2 * VN.X + .B / 2 * VC.X
P(12).Y = .Y2 - .h / 2 * VN.Y + .B / 2 * VC.Y
P(12).Z = .Z2 - .h / 2 * VN.Z + .B / 2 * VC.Z


Select Case EstructuraVis.ColBarra.Item(BarraEnCurs).Material

  Case "Formigó"
  'ALA SUPERIOR
  m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 1) = CreateLitVertex(P(1).X,
P(1).Y, P(1).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
  m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 2) = CreateLitVertex(P(7).X,
P(7).Y, P(7).Z, RGB(200, 200, 200), RGB(175, 175, 175), 0, 2 * .L)
  m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 3) = CreateLitVertex(P(9).X,
P(9).Y, P(9).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

  m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 4) = CreateLitVertex(P(1).X,
P(1).Y, P(1).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
  m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 5) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(200, 200, 200), RGB(200, 200, 200), .B, 0)
  m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 6) = CreateLitVertex(P(9).X,
P(9).Y, P(9).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

  'ANIMA
  m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 7) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
  m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 8) = CreateLitVertex(P(8).X,
P(8).Y, P(8).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 2 * .L)
  m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 9) = CreateLitVertex(P(11).X,
P(11).Y, P(11).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h, 2 * .L)

  m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 10) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
  m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 11) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h, 0)
  m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 12) = CreateLitVertex(P(11).X,
P(11).Y, P(11).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h, 2 * .L)

  'ALA INFERIOR
  m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 13) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
  m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 14) = CreateLitVertex(P(10).X,
P(10).Y, P(10).Z, RGB(200, 200, 200), RGB(175, 175, 175), 0, 2 * .L)
  m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 15) = CreateLitVertex(P(12).X,
P(12).Y, P(12).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

  m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 16) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
```

```
        m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 17) = CreateLitVertex(P(6).X,
P(6).Y, P(6).Z, RGB(200, 200, 200), RGB(200, 200, 200), .B, 0)
        m_VertexsDePissarra2((BarraEnCurs - 1) * 18 + 18) = CreateLitVertex(P(12).X,
P(12).Y, P(12).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)


        Case "Acer"

        'ALA SUPERIOR
        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 1) = CreateLitVertex(P(1).X,
P(1).Y, P(1).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 2) = CreateLitVertex(P(7).X,
P(7).Y, P(7).Z, RGB(200, 200, 200), RGB(175, 175, 175), 0, 2 * .L)
        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 3) = CreateLitVertex(P(9).X,
P(9).Y, P(9).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 4) = CreateLitVertex(P(1).X,
P(1).Y, P(1).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 5) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(200, 200, 200), RGB(200, 200, 200), .B, 0)
        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 6) = CreateLitVertex(P(9).X,
P(9).Y, P(9).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

        'ANIMA
        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 7) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 8) = CreateLitVertex(P(8).X,
P(8).Y, P(8).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 2 * .L)
        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 9) = CreateLitVertex(P(11).X,
P(11).Y, P(11).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h, 2 * .L)

        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 10) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 11) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h, 0)
        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 12) = CreateLitVertex(P(11).X,
P(11).Y, P(11).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h, 2 * .L)

        'ALA INFERIOR
        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 13) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 14) = CreateLitVertex(P(10).X,
P(10).Y, P(10).Z, RGB(200, 200, 200), RGB(175, 175, 175), 0, 2 * .L)
        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 15) = CreateLitVertex(P(12).X,
P(12).Y, P(12).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 16) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 17) = CreateLitVertex(P(6).X,
P(6).Y, P(6).Z, RGB(200, 200, 200), RGB(200, 200, 200), .B, 0)
```

```
        m_VertexsDePissarraS2((BarraEnCurs - 1) * 18 + 18) = CreateLitVertex(P(12).X,
P(12).Y, P(12).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)


        Case Else ' Qualsevol altre material

        'ALA SUPERIOR
        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 1) = CreateLitVertex(P(1).X,
P(1).Y, P(1).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 2) = CreateLitVertex(P(7).X,
P(7).Y, P(7).Z, RGB(200, 200, 200), RGB(175, 175, 175), 0, 4 * .L)
        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 3) = CreateLitVertex(P(9).X,
P(9).Y, P(9).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B * 2, 4 * .L)

        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 4) = CreateLitVertex(P(1).X,
P(1).Y, P(1).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 5) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(200, 200, 200), RGB(200, 200, 200), .B * 2, 0)
        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 6) = CreateLitVertex(P(9).X,
P(9).Y, P(9).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B * 2, 4 * .L)

        'ANIMA
        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 7) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 8) = CreateLitVertex(P(8).X,
P(8).Y, P(8).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 4 * .L)
        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 9) = CreateLitVertex(P(11).X,
P(11).Y, P(11).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h * 2, 2 * .L)

        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 10) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 11) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h * 2, 0)
        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 12) = CreateLitVertex(P(11).X,
P(11).Y, P(11).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h * 2, 4 * .L)

        'ALA INFERIOR
        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 13) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 14) = CreateLitVertex(P(10).X,
P(10).Y, P(10).Z, RGB(200, 200, 200), RGB(175, 175, 175), 0, 4 * .L)
        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 15) = CreateLitVertex(P(12).X,
P(12).Y, P(12).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B * 2, 4 * .L)

        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 16) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 17) = CreateLitVertex(P(6).X,
P(6).Y, P(6).Z, RGB(200, 200, 200), RGB(200, 200, 200), .B * 2, 0)
        m_VertexsDePissarraA2((BarraEnCurs - 1) * 18 + 18) = CreateLitVertex(P(12).X,
P(12).Y, P(12).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B * 2, 4 * .L)
```

```
        End Select
        End With

      Case 4
      DMdlDirectX.AssignaGeometriaPissarraPerfilUPN
      Case 5
      DMdlDirectX.AssignaGeometriaPissarraPerfilL
      Case 9


   End Select


   Next BarraEnCurs

        Set EstructuraVis.DadesDirectX.vbufferpissarra =
EstructuraVis.DadesDirectX.D3DDevicePissarra.CreateVertexBuffer(Len(m_VertexsDePiss
arra(1)) * MaxBarra * 36, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.vbufferpissarra, 0,
Len(m_VertexsDePissarra(1)) * MaxBarra * 36, 0, m_VertexsDePissarra(1)

        Set EstructuraVis.DadesDirectX.VBufferPissarraS =
EstructuraVis.DadesDirectX.D3DDevicePissarra.CreateVertexBuffer(Len(m_VertexsDePiss
arraS(1)) * MaxBarra * 36, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferPissarraS, 0,
Len(m_VertexsDePissarraS(1)) * MaxBarra * 36, 0, m_VertexsDePissarraS(1)

        Set EstructuraVis.DadesDirectX.VBufferPissarraA =
EstructuraVis.DadesDirectX.D3DDevicePissarra.CreateVertexBuffer(Len(m_VertexsDePiss
arraA(1)) * MaxBarra * 36, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferPissarraA, 0,
Len(m_VertexsDePissarraA(1)) * MaxBarra * 36, 0, m_VertexsDePissarraA(1)



        Set EstructuraVis.DadesDirectX.VBufferPissarra2 =
EstructuraVis.DadesDirectX.D3DDevicePissarra.CreateVertexBuffer(Len(m_VertexsDePiss
arra2(1)) * MaxBarra * 36, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferPissarra2, 0,
Len(m_VertexsDePissarra2(1)) * MaxBarra * 18, 0, m_VertexsDePissarra2(1)

        Set EstructuraVis.DadesDirectX.VBufferPissarraS2 =
EstructuraVis.DadesDirectX.D3DDevicePissarra.CreateVertexBuffer(Len(m_VertexsDePiss
arraS2(1)) * MaxBarra * 36, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferPissarraS2, 0,
Len(m_VertexsDePissarraS2(1)) * MaxBarra * 18, 0, m_VertexsDePissarraS2(1)
```

```
        Set EstructuraVis.DadesDirectX.VBufferPissarraA2 =
EstructuraVis.DadesDirectX.D3DDevicePissarra.CreateVertexBuffer(Len(m_VertexsDePiss
arraA2(1)) * MaxBarra * 36, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferPissarraA2, 0,
Len(m_VertexsDePissarraA2(1)) * MaxBarra * 18, 0, m_VertexsDePissarraA2(1)



        Set EstructuraVis.DadesDirectX.VBufferPissarra3 =
EstructuraVis.DadesDirectX.D3DDevicePissarra.CreateVertexBuffer(Len(m_VertexsDePiss
arra3(1)) * MaxBarra * 240, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferPissarra3, 0,
Len(m_VertexsDePissarra3(1)) * MaxBarra * 240, 0, m_VertexsDePissarra3(1)

        Set EstructuraVis.DadesDirectX.VBufferPissarraS3 =
EstructuraVis.DadesDirectX.D3DDevicePissarra.CreateVertexBuffer(Len(m_VertexsDePiss
arraS3(1)) * MaxBarra * 240, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferPissarraS3, 0,
Len(m_VertexsDePissarraS3(1)) * MaxBarra * 240, 0, m_VertexsDePissarraS3(1)

        Set EstructuraVis.DadesDirectX.VBufferPissarraA3 =
EstructuraVis.DadesDirectX.D3DDevicePissarra.CreateVertexBuffer(Len(m_VertexsDePiss
arraA3(1)) * MaxBarra * 240, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferPissarraA3, 0,
Len(m_VertexsDePissarraA3(1)) * MaxBarra * 240, 0, m_VertexsDePissarraA3(1)



        Set EstructuraVis.DadesDirectX.VBufferPissarra4 =
EstructuraVis.DadesDirectX.D3DDevicePissarra.CreateVertexBuffer(Len(m_VertexsDePiss
arra4(1)) * MaxBarra * 18, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferPissarra4, 0,
Len(m_VertexsDePissarra4(1)) * MaxBarra * 18, 0, m_VertexsDePissarra4(1)

        Set EstructuraVis.DadesDirectX.VBufferPissarraS4 =
EstructuraVis.DadesDirectX.D3DDevicePissarra.CreateVertexBuffer(Len(m_VertexsDePiss
arraS4(1)) * MaxBarra * 18, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferPissarraS4, 0,
Len(m_VertexsDePissarraS4(1)) * MaxBarra * 18, 0, m_VertexsDePissarraS4(1)

        Set EstructuraVis.DadesDirectX.VBufferPissarraA4 =
EstructuraVis.DadesDirectX.D3DDevicePissarra.CreateVertexBuffer(Len(m_VertexsDePiss
arraA4(1)) * MaxBarra * 18, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferPissarraA4, 0,
Len(m_VertexsDePissarraA4(1)) * MaxBarra * 18, 0, m_VertexsDePissarraA4(1)



        Set EstructuraVis.DadesDirectX.VBufferPissarra5 =
EstructuraVis.DadesDirectX.D3DDevicePissarra.CreateVertexBuffer(Len(m_VertexsDePiss
arra5(1)) * MaxBarra * 12, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferPissarra5, 0,
Len(m_VertexsDePissarra5(1)) * MaxBarra * 12, 0, m_VertexsDePissarra5(1)
```

```
        Set EstructuraVis.DadesDirectX.VBufferPissarraS5 =
EstructuraVis.DadesDirectX.D3DDevicePissarra.CreateVertexBuffer(Len(m_VertexsDePiss
arraS5(1)) * MaxBarra * 12, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferPissarraS5, 0,
Len(m_VertexsDePissarraS5(1)) * MaxBarra * 12, 0, m_VertexsDePissarraS5(1)

        Set EstructuraVis.DadesDirectX.VBufferPissarraA5 =
EstructuraVis.DadesDirectX.D3DDevicePissarra.CreateVertexBuffer(Len(m_VertexsDePiss
arraA5(1)) * MaxBarra * 12, 0, TL_FVF, D3DPOOL_DEFAULT)
        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferPissarraA5, 0,
Len(m_VertexsDePissarraA5(1)) * MaxBarra * 12, 0, m_VertexsDePissarraA5(1)



End Sub



Sub AssignaGeometriaPissarraPerfilUPN()
Dim BarraEnCurs As Long
Dim MaxBarra As Long
Dim VDirector(1 To 3) As Double, VD As D3DVECTOR
Dim VOrto(1 To 3) As Double, VC As D3DVECTOR
Dim VAux(1 To 3) As Double, VN As D3DVECTOR
Dim Modul As Double
Dim VectorAuxiliar(100) As Double
Dim P(1 To 12) As D3DVECTOR
Dim PC(1 To 40) As D3DVECTOR
If EstructuraVis.ColBarra.Count = 0 Then Exit Sub
Dim i As Long
Dim Color As Long, Ample As Single


MaxBarra = EstructuraVis.ColBarra.Count

ReDim m_VertexsDePissarra4(1 To 18 * MaxBarra)
ReDim m_VertexsDePissarraS4(1 To 18 * MaxBarra)
ReDim m_VertexsDePissarraA4(1 To 18 * MaxBarra)



For BarraEnCurs = 1 To MaxBarra

Select Case EstructuraVis.ColBarra.Item(BarraEnCurs).TipusPerfil
   Case 4

      With EstructuraVis.ColBarra.Item(BarraEnCurs)

      VD = CreaVector(.X2 - .X1, .Y2 - .Y1, .Z2 - .Z1)
      D3DXVec3Normalize VD, VD
```

```
VC = CreaVector(.VxOrto, .VyOrto, .VzOrto)
D3DXVec3Normalize VC, VC

D3DXVec3Cross VN, VD, VC
D3DXVec3Normalize VN, VN

P(3).X = .X1 + .h / 2 * VN.X - .DE * 10 * VC.X
P(3).Y = .Y1 + .h / 2 * VN.Y - .DE * 10 * VC.Y
P(3).Z = .Z1 + .h / 2 * VN.Z - .DE * 10 * VC.Z

P(4).X = .X1 - .h / 2 * VN.X - .DE * 10 * VC.X
P(4).Y = .Y1 - .h / 2 * VN.Y - .DE * 10 * VC.Y
P(4).Z = .Z1 - .h / 2 * VN.Z - .DE * 10 * VC.Z

P(1).X = .X1 + .h / 2 * VN.X + (.B - .DE * 10) * VC.X
P(1).Y = .Y1 + .h / 2 * VN.Y + (.B - .DE * 10) * VC.Y
P(1).Z = .Z1 + .h / 2 * VN.Z + (.B - .DE * 10) * VC.Z

P(2).X = .X1 - .h / 2 * VN.X + (.B - .DE * 10) * VC.X
P(2).Y = .Y1 - .h / 2 * VN.Y + (.B - .DE * 10) * VC.Y
P(2).Z = .Z1 - .h / 2 * VN.Z + (.B - .DE * 10) * VC.Z

P(7).X = .X2 + .h / 2 * VN.X - .DE * 10 * VC.X
P(7).Y = .Y2 + .h / 2 * VN.Y - .DE * 10 * VC.Y
P(7).Z = .Z2 + .h / 2 * VN.Z - .DE * 10 * VC.Z

P(8).X = .X2 - .h / 2 * VN.X - .DE * 10 * VC.X
P(8).Y = .Y2 - .h / 2 * VN.Y - .DE * 10 * VC.Y
P(8).Z = .Z2 - .h / 2 * VN.Z - .DE * 10 * VC.Z

P(5).X = .X2 + .h / 2 * VN.X + (.B - .DE * 10) * VC.X
P(5).Y = .Y2 + .h / 2 * VN.Y + (.B - .DE * 10) * VC.Y
P(5).Z = .Z2 + .h / 2 * VN.Z + (.B - .DE * 10) * VC.Z

P(6).X = .X2 - .h / 2 * VN.X + (.B - .DE * 10) * VC.X
P(6).Y = .Y2 - .h / 2 * VN.Y + (.B - .DE * 10) * VC.Y
P(6).Z = .Z2 - .h / 2 * VN.Z + (.B - .DE * 10) * VC.Z


Select Case EstructuraVis.ColBarra.Item(BarraEnCurs).Material

  Case "Formigó"
  'ALA SUPERIOR
  m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 1) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
  m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 2) = CreateLitVertex(P(7).X,
P(7).Y, P(7).Z, RGB(200, 200, 200), RGB(175, 175, 175), 0, 2 * .L)
  m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 3) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)
```

```
        m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 4) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 5) = CreateLitVertex(P(1).X,
P(1).Y, P(1).Z, RGB(200, 200, 200), RGB(200, 200, 200), .B, 0)
        m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 6) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

        'ANIMA
        m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 7) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
        m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 8) = CreateLitVertex(P(7).X,
P(7).Y, P(7).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 2 * .L)
        m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 9) = CreateLitVertex(P(8).X,
P(8).Y, P(8).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h, 2 * .L)

        m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 10) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
        m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 11) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h, 0)
        m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 12) = CreateLitVertex(P(8).X,
P(8).Y, P(8).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h, 2 * .L)

        'ALA INFERIOR
        m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 13) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 14) = CreateLitVertex(P(8).X,
P(8).Y, P(8).Z, RGB(200, 200, 200), RGB(175, 175, 175), 0, 2 * .L)
        m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 15) = CreateLitVertex(P(6).X,
P(6).Y, P(6).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

        m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 16) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 17) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(200, 200, 200), RGB(200, 200, 200), .B, 0)
        m_VertexsDePissarra4((BarraEnCurs - 1) * 18 + 18) = CreateLitVertex(P(6).X,
P(6).Y, P(6).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

        Case "Acer"
        'ALA SUPERIOR
        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 1) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 2) = CreateLitVertex(P(7).X,
P(7).Y, P(7).Z, RGB(200, 200, 200), RGB(175, 175, 175), 0, 2 * .L)
        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 3) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 4) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 5) = CreateLitVertex(P(1).X,
P(1).Y, P(1).Z, RGB(200, 200, 200), RGB(200, 200, 200), .B, 0)
```

```
        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 6) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

        'ANIMA
        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 7) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 8) = CreateLitVertex(P(7).X,
P(7).Y, P(7).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 2 * .L)
        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 9) = CreateLitVertex(P(8).X,
P(8).Y, P(8).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h, 2 * .L)

        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 10) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 11) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h, 0)
        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 12) = CreateLitVertex(P(8).X,
P(8).Y, P(8).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h, 2 * .L)

        'ALA INFERIOR
        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 13) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 14) = CreateLitVertex(P(8).X,
P(8).Y, P(8).Z, RGB(200, 200, 200), RGB(175, 175, 175), 0, 2 * .L)
        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 15) = CreateLitVertex(P(6).X,
P(6).Y, P(6).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 16) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 17) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(200, 200, 200), RGB(200, 200, 200), .B, 0)
        m_VertexsDePissarraS4((BarraEnCurs - 1) * 18 + 18) = CreateLitVertex(P(6).X,
P(6).Y, P(6).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

        Case Else
        'ALA SUPERIOR
        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 1) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 2) = CreateLitVertex(P(7).X,
P(7).Y, P(7).Z, RGB(200, 200, 200), RGB(175, 175, 175), 0, 2 * .L)
        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 3) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 4) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 5) = CreateLitVertex(P(1).X,
P(1).Y, P(1).Z, RGB(200, 200, 200), RGB(200, 200, 200), .B, 0)
        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 6) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

        'ANIMA
```

```
        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 7) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 8) = CreateLitVertex(P(7).X,
P(7).Y, P(7).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 2 * .L)
        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 9) = CreateLitVertex(P(8).X,
P(8).Y, P(8).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h, 2 * .L)

        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 10) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 11) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h, 0)
        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 12) = CreateLitVertex(P(8).X,
P(8).Y, P(8).Z, RGB(150, 150, 150), RGB(150, 150, 150), .h, 2 * .L)

        'ALA INFERIOR
        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 13) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 14) = CreateLitVertex(P(8).X,
P(8).Y, P(8).Z, RGB(200, 200, 200), RGB(175, 175, 175), 0, 2 * .L)
        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 15) = CreateLitVertex(P(6).X,
P(6).Y, P(6).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 16) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 17) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(200, 200, 200), RGB(200, 200, 200), .B, 0)
        m_VertexsDePissarraA4((BarraEnCurs - 1) * 18 + 18) = CreateLitVertex(P(6).X,
P(6).Y, P(6).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)


    End Select
    End With
End Select
Next BarraEnCurs



End Sub


Sub AssignaGeometriaPissarraPerfilL()
Dim BarraEnCurs As Long
Dim MaxBarra As Long
Dim VDirector(1 To 3) As Double, VD As D3DVECTOR
Dim VOrto(1 To 3) As Double, VC As D3DVECTOR
Dim VAux(1 To 3) As Double, VN As D3DVECTOR
Dim Modul As Double
Dim VectorAuxiliar(100) As Double
Dim P(1 To 12) As D3DVECTOR
Dim PC(1 To 40) As D3DVECTOR
If EstructuraVis.ColBarra.Count = 0 Then Exit Sub
```

```
Dim i As Long
Dim Color As Long, Ample As Single


MaxBarra = EstructuraVis.ColBarra.Count

ReDim m_VertexsDePissarra5(1 To 12 * MaxBarra)
ReDim m_VertexsDePissarraS5(1 To 12 * MaxBarra)
ReDim m_VertexsDePissarraA5(1 To 12 * MaxBarra)



For BarraEnCurs = 1 To MaxBarra

Select Case EstructuraVis.ColBarra.Item(BarraEnCurs).TipusPerfil
   Case 5

      With EstructuraVis.ColBarra.Item(BarraEnCurs)

      VD = CreaVector(.X2 - .X1, .Y2 - .Y1, .Z2 - .Z1)
      D3DXVec3Normalize VD, VD

      VC = CreaVector(.VxOrto, .VyOrto, .VzOrto)
      D3DXVec3Normalize VC, VC

      D3DXVec3Cross VN, VD, VC
      D3DXVec3Normalize VN, VN

      P(3).X = .X1 + (-.h * 10) * VN.X - (.B - .h * 10) * VC.X
      P(3).Y = .Y1 + (-.h * 10) * VN.Y - (.B - .h * 10) * VC.Y
      P(3).Z = .Z1 + (-.h * 10) * VN.Z - (.B - .h * 10) * VC.Z

      P(1).X = .X1 + (.B - .h * 10) * VN.X - (-.h * 10) * VC.X
      P(1).Y = .Y1 + (.B - .h * 10) * VN.Y - (-.h * 10) * VC.Y
      P(1).Z = .Z1 + (.B - .h * 10) * VN.Z - (-.h * 10) * VC.Z

      P(2).X = .X1 + (-.h * 10) * VN.X - (-.h * 10) * VC.X
      P(2).Y = .Y1 + (-.h * 10) * VN.Y - (-.h * 10) * VC.Y
      P(2).Z = .Z1 + (-.h * 10) * VN.Z - (-.h * 10) * VC.Z

      P(6).X = .X2 + (-.h * 10) * VN.X - (.B - .h * 10) * VC.X
      P(6).Y = .Y2 + (-.h * 10) * VN.Y - (.B - .h * 10) * VC.Y
      P(6).Z = .Z2 + (-.h * 10) * VN.Z - (.B - .h * 10) * VC.Z

      P(4).X = .X2 + (.B - .h * 10) * VN.X - (-.h * 10) * VC.X
      P(4).Y = .Y2 + (.B - .h * 10) * VN.Y - (-.h * 10) * VC.Y
      P(4).Z = .Z2 + (.B - .h * 10) * VN.Z - (-.h * 10) * VC.Z

      P(5).X = .X2 + (-.h * 10) * VN.X - (-.h * 10) * VC.X
      P(5).Y = .Y2 + (-.h * 10) * VN.Y - (-.h * 10) * VC.Y
      P(5).Z = .Z2 + (-.h * 10) * VN.Z - (-.h * 10) * VC.Z
```

```
Select Case EstructuraVis.ColBarra.Item(BarraEnCurs).Material

    Case "Formigó"

    'ANIMA
    m_VertexsDePissarra5((BarraEnCurs - 1) * 12 + 1) = CreateLitVertex(P(1).X,
P(1).Y, P(1).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
    m_VertexsDePissarra5((BarraEnCurs - 1) * 12 + 2) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 2 * .L)
    m_VertexsDePissarra5((BarraEnCurs - 1) * 12 + 3) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(150, 150, 150), RGB(150, 150, 150), .B, 2 * .L)

    m_VertexsDePissarra5((BarraEnCurs - 1) * 12 + 4) = CreateLitVertex(P(1).X,
P(1).Y, P(1).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
    m_VertexsDePissarra5((BarraEnCurs - 1) * 12 + 5) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(150, 150, 150), RGB(150, 150, 150), .B, 0)
    m_VertexsDePissarra5((BarraEnCurs - 1) * 12 + 6) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(150, 150, 150), RGB(150, 150, 150), .B, 2 * .L)

    'ALA INFERIOR
    m_VertexsDePissarra5((BarraEnCurs - 1) * 12 + 7) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
    m_VertexsDePissarra5((BarraEnCurs - 1) * 12 + 8) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(200, 200, 200), RGB(175, 175, 175), 0, 2 * .L)
    m_VertexsDePissarra5((BarraEnCurs - 1) * 12 + 9) = CreateLitVertex(P(6).X,
P(6).Y, P(6).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

    m_VertexsDePissarra5((BarraEnCurs - 1) * 12 + 10) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
    m_VertexsDePissarra5((BarraEnCurs - 1) * 12 + 11) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(200, 200, 200), RGB(200, 200, 200), .B, 0)
    m_VertexsDePissarra5((BarraEnCurs - 1) * 12 + 12) = CreateLitVertex(P(6).X,
P(6).Y, P(6).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

    Case "Acer"

     'ANIMA
    m_VertexsDePissarraS5((BarraEnCurs - 1) * 12 + 1) = CreateLitVertex(P(1).X,
P(1).Y, P(1).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
    m_VertexsDePissarraS5((BarraEnCurs - 1) * 12 + 2) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 2 * .L)
    m_VertexsDePissarraS5((BarraEnCurs - 1) * 12 + 3) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(150, 150, 150), RGB(150, 150, 150), .B, 2 * .L)

    m_VertexsDePissarraS5((BarraEnCurs - 1) * 12 + 4) = CreateLitVertex(P(1).X,
P(1).Y, P(1).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
    m_VertexsDePissarraS5((BarraEnCurs - 1) * 12 + 5) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(150, 150, 150), RGB(150, 150, 150), .B, 0)
```

```
        m_VertexsDePissarraS5((BarraEnCurs - 1) * 12 + 6) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(150, 150, 150), RGB(150, 150, 150), .B, 2 * .L)

        'ALA INFERIOR
        m_VertexsDePissarraS5((BarraEnCurs - 1) * 12 + 7) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraS5((BarraEnCurs - 1) * 12 + 8) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(200, 200, 200), RGB(175, 175, 175), 0, 2 * .L)
        m_VertexsDePissarraS5((BarraEnCurs - 1) * 12 + 9) = CreateLitVertex(P(6).X,
P(6).Y, P(6).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

        m_VertexsDePissarraS5((BarraEnCurs - 1) * 12 + 10) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraS5((BarraEnCurs - 1) * 12 + 11) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(200, 200, 200), RGB(200, 200, 200), .B, 0)
        m_VertexsDePissarraS5((BarraEnCurs - 1) * 12 + 12) = CreateLitVertex(P(6).X,
P(6).Y, P(6).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

        Case Else

        m_VertexsDePissarraA5((BarraEnCurs - 1) * 12 + 1) = CreateLitVertex(P(1).X,
P(1).Y, P(1).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
        m_VertexsDePissarraA5((BarraEnCurs - 1) * 12 + 2) = CreateLitVertex(P(4).X,
P(4).Y, P(4).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 2 * .L)
        m_VertexsDePissarraA5((BarraEnCurs - 1) * 12 + 3) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(150, 150, 150), RGB(150, 150, 150), .B, 2 * .L)

        m_VertexsDePissarraA5((BarraEnCurs - 1) * 12 + 4) = CreateLitVertex(P(1).X,
P(1).Y, P(1).Z, RGB(150, 150, 150), RGB(150, 150, 150), 0, 0)
        m_VertexsDePissarraA5((BarraEnCurs - 1) * 12 + 5) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(150, 150, 150), RGB(150, 150, 150), .B, 0)
        m_VertexsDePissarraA5((BarraEnCurs - 1) * 12 + 6) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(150, 150, 150), RGB(150, 150, 150), .B, 2 * .L)

        'ALA INFERIOR
        m_VertexsDePissarraA5((BarraEnCurs - 1) * 12 + 7) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraA5((BarraEnCurs - 1) * 12 + 8) = CreateLitVertex(P(5).X,
P(5).Y, P(5).Z, RGB(200, 200, 200), RGB(175, 175, 175), 0, 2 * .L)
        m_VertexsDePissarraA5((BarraEnCurs - 1) * 12 + 9) = CreateLitVertex(P(6).X,
P(6).Y, P(6).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)

        m_VertexsDePissarraA5((BarraEnCurs - 1) * 12 + 10) = CreateLitVertex(P(2).X,
P(2).Y, P(2).Z, RGB(250, 250, 250), RGB(250, 250, 250), 0, 0)
        m_VertexsDePissarraA5((BarraEnCurs - 1) * 12 + 11) = CreateLitVertex(P(3).X,
P(3).Y, P(3).Z, RGB(200, 200, 200), RGB(200, 200, 200), .B, 0)
        m_VertexsDePissarraA5((BarraEnCurs - 1) * 12 + 12) = CreateLitVertex(P(6).X,
P(6).Y, P(6).Z, RGB(175, 175, 175), RGB(175, 175, 175), .B, 2 * .L)


    End Select
```

```
    End With
End Select
Next BarraEnCurs



End Sub

Sub AssignaGeometriaEixos()

Dim TamanyEixos As Double
Dim LRdC As Double 'Longitud Relativa del Cap
Dim GRdC As Double 'Gruix Relatiu del Cap
Dim V1 As D3DVECTOR
Dim V2 As D3DVECTOR
Dim V3 As D3DVECTOR
Dim Modul As Double
Dim Alfa As Double, Beta As Double, Dist As Double


'Determino la longitud dels eixos, proporcional a la distància de la càmera _
al 0, 0, 0
Set EstructuraVis.DadesDirectX.VBufferEixosV1 = Nothing
Set EstructuraVis.DadesDirectX.VBufferEixosV2 = Nothing
Set EstructuraVis.DadesDirectX.VBufferEixosV3 = Nothing
Set EstructuraVis.DadesDirectX.VBufferEixosV4 = Nothing

Set EstructuraVis.DadesDirectX.VBufferDeCapsDeEixosV1 = Nothing
Set EstructuraVis.DadesDirectX.VBufferDeCapsDeEixosV2 = Nothing
Set EstructuraVis.DadesDirectX.VBufferDeCapsDeEixosV3 = Nothing
Set EstructuraVis.DadesDirectX.VBufferDeCapsDeEixosV4 = Nothing


Modul = DMdlDirectX.DistanciaDelCentreDeGravetatALaCameraALaVistaEnCurs

TamanyEixos = EstructuraVis.Preferencies.TamanyEixos * Modul

'Primer dibuixo els eixos
LRdC = EstructuraVis.Preferencies.TamanyRelatiuCapEixos
'LRdC = LRdC * TamanyEixos
GRdC = EstructuraVis.Preferencies.GruixRelatiuCapEixos
GRdC = GRdC * TamanyEixos


m_VertexsDeEixos(1) = CreateLitVertex(0, 0, 0, EstructuraVis.Preferencies.ColorEixX,
EstructuraVis.Preferencies.ColorEixX, 0, 0)
m_VertexsDeEixos(2) = CreateLitVertex((TamanyEixos) * 0.9, 0, 0,
EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0, 0)
m_VertexsDeEixos(3) = CreateLitVertex(0, 0, 0, EstructuraVis.Preferencies.ColorEixY,
EstructuraVis.Preferencies.ColorEixY, 0, 0)
```

```
m_VertexsDeEixos(4) = CreateLitVertex(0, (TamanyEixos) * 0.9, 0,
EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0, 0)
m_VertexsDeEixos(5) = CreateLitVertex(0, 0, 0, EstructuraVis.Preferencies.ColorEixZ,
EstructuraVis.Preferencies.ColorEixZ, 0, 0)
m_VertexsDeEixos(6) = CreateLitVertex(0, 0, (TamanyEixos) * 0.9,
EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0, 0)



'Cal Fer aquesta assignació per a cada finestra

    Set EstructuraVis.DadesDirectX.VBufferEixosV3 =
EstructuraVis.DadesDirectX.D3DDeviceV3.CreateVertexBuffer _
 (Len(m_VertexsDeEixos(1)) * 6, 0, TL_FVF, D3DPOOL_DEFAULT)
    'Set EstructuraVis.DadesDirectX.VBufferEixosV2 =
EstructuraVis.DadesDirectX.D3DDeviceV2.CreateVertexBuffer _
 (Len(m_VertexsDeEixos(1)) * 6, 0, TL_FVF, D3DPOOL_DEFAULT)
    'Set EstructuraVis.DadesDirectX.VBufferEixosV1 =
EstructuraVis.DadesDirectX.D3DDeviceV1.CreateVertexBuffer _
 (Len(m_VertexsDeEixos(1)) * 6, 0, TL_FVF, D3DPOOL_DEFAULT)
    'Set EstructuraVis.DadesDirectX.VBufferEixosV4 =
EstructuraVis.DadesDirectX.D3DDeviceV4.CreateVertexBuffer _
 (Len(m_VertexsDeEixos(1)) * 6, 0, TL_FVF, D3DPOOL_DEFAULT)

'I fill the created vertex with the data

        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferEixosV3, 0, _
        Len(m_VertexsDeEixos(1)) * 6, 0, m_VertexsDeEixos(1)

        'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferEixosV2, 0, _
        Len(m_VertexsDeEixos(1)) * 6, 0, m_VertexsDeEixos(1)

        'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferEixosV1, 0, _
        Len(m_VertexsDeEixos(1)) * 6, 0, m_VertexsDeEixos(1)

        'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferEixosV4, 0, _
        Len(m_VertexsDeEixos(1)) * 6, 0, m_VertexsDeEixos(1)


'Ara dibuixo els caps


'Ara dibuixo els caps

'Cap X
m_VertexsDeCapsDeEixos(1) = CreateLitVertex((TamanyEixos), 0, 0,
EstructuraVis.Preferencies.ColorEixXCap, EstructuraVis.Preferencies.ColorEixXCap, 0, 0)
m_VertexsDeCapsDeEixos(3) = CreateLitVertex((TamanyEixos) * (1 - LRdC), (GRdC) * 3,
(GRdC), EstructuraVis.Preferencies.ColorEixXCap,
EstructuraVis.Preferencies.ColorEixXCap, 0, 0)
```

```
m_VertexsDeCapsDeEixos(2) = CreateLitVertex((TamanyEixos) * (1 - LRdC), (GRdC) * 3, -
(GRdC), EstructuraVis.Preferencies.ColorEixXCap,
EstructuraVis.Preferencies.ColorEixXCap, 0, 0)

m_VertexsDeCapsDeEixos(4) = CreateLitVertex((TamanyEixos), 0, 0,
EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0, 0)
m_VertexsDeCapsDeEixos(5) = CreateLitVertex((TamanyEixos) * (1 - LRdC), (GRdC) * 3,
(GRdC), EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0, 0)
m_VertexsDeCapsDeEixos(6) = CreateLitVertex((TamanyEixos) * (1 - LRdC), (GRdC),
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0,
0)

m_VertexsDeCapsDeEixos(7) = CreateLitVertex((TamanyEixos), 0, 0,
EstructuraVis.Preferencies.ColorEixXCap, EstructuraVis.Preferencies.ColorEixXCap, 0, 0)
m_VertexsDeCapsDeEixos(8) = CreateLitVertex((TamanyEixos) * (1 - LRdC), -(GRdC),
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixXCap,
EstructuraVis.Preferencies.ColorEixXCap, 0, 0)
m_VertexsDeCapsDeEixos(9) = CreateLitVertex((TamanyEixos) * (1 - LRdC), (GRdC),
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixXCap,
EstructuraVis.Preferencies.ColorEixXCap, 0, 0)

m_VertexsDeCapsDeEixos(10) = CreateLitVertex((TamanyEixos), 0, 0,
EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0, 0)
m_VertexsDeCapsDeEixos(11) = CreateLitVertex((TamanyEixos) * (1 - LRdC), -(GRdC),
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0,
0)
m_VertexsDeCapsDeEixos(12) = CreateLitVertex((TamanyEixos) * (1 - LRdC), -(GRdC) *
3, (GRdC), EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0,
0)

m_VertexsDeCapsDeEixos(13) = CreateLitVertex((TamanyEixos), 0, 0,
EstructuraVis.Preferencies.ColorEixXCap, EstructuraVis.Preferencies.ColorEixXCap, 0, 0)
m_VertexsDeCapsDeEixos(14) = CreateLitVertex((TamanyEixos) * (1 - LRdC), -(GRdC) *
3, -(GRdC), EstructuraVis.Preferencies.ColorEixXCap,
EstructuraVis.Preferencies.ColorEixXCap, 0, 0)
m_VertexsDeCapsDeEixos(15) = CreateLitVertex((TamanyEixos) * (1 - LRdC), -(GRdC) *
3, (GRdC), EstructuraVis.Preferencies.ColorEixXCap,
EstructuraVis.Preferencies.ColorEixXCap, 0, 0)

m_VertexsDeCapsDeEixos(16) = CreateLitVertex((TamanyEixos), 0, 0,
EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0, 0)
m_VertexsDeCapsDeEixos(17) = CreateLitVertex((TamanyEixos) * (1 - LRdC), -(GRdC) *
3, -(GRdC), EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0,
0)
m_VertexsDeCapsDeEixos(18) = CreateLitVertex((TamanyEixos) * (1 - LRdC), -(GRdC), -
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0,
0)

m_VertexsDeCapsDeEixos(19) = CreateLitVertex((TamanyEixos), 0, 0,
EstructuraVis.Preferencies.ColorEixXCap, EstructuraVis.Preferencies.ColorEixXCap, 0, 0)
```

```
m_VertexsDeCapsDeEixos(20) = CreateLitVertex((TamanyEixos) * (1 - LRdC), (GRdC), -
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixXCap,
EstructuraVis.Preferencies.ColorEixXCap, 0, 0)
m_VertexsDeCapsDeEixos(21) = CreateLitVertex((TamanyEixos) * (1 - LRdC), -(GRdC), -
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixXCap,
EstructuraVis.Preferencies.ColorEixXCap, 0, 0)

m_VertexsDeCapsDeEixos(22) = CreateLitVertex((TamanyEixos), 0, 0,
EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0, 0)
m_VertexsDeCapsDeEixos(23) = CreateLitVertex((TamanyEixos) * (1 - LRdC), (GRdC) * 3,
-(GRdC), EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0, 0)
m_VertexsDeCapsDeEixos(24) = CreateLitVertex((TamanyEixos) * (1 - LRdC), (GRdC), -
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0,
0)



m_VertexsDeCapsDeEixos(46) = CreateLitVertex((TamanyEixos) * (1 - LRdC), 0, 0,
EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0, 0)
m_VertexsDeCapsDeEixos(47) = CreateLitVertex((TamanyEixos) * (1 - LRdC), (GRdC) * 3,
(GRdC), EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0, 0)
m_VertexsDeCapsDeEixos(48) = CreateLitVertex((TamanyEixos) * (1 - LRdC), (GRdC) * 3,
-(GRdC), EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0, 0)

m_VertexsDeCapsDeEixos(25) = CreateLitVertex((TamanyEixos) * (1 - LRdC), 0, 0,
EstructuraVis.Preferencies.ColorEixXCap, EstructuraVis.Preferencies.ColorEixXCap, 0, 0)
m_VertexsDeCapsDeEixos(26) = CreateLitVertex((TamanyEixos) * (1 - LRdC), (GRdC) * 3,
(GRdC), EstructuraVis.Preferencies.ColorEixXCap,
EstructuraVis.Preferencies.ColorEixXCap, 0, 0)
m_VertexsDeCapsDeEixos(27) = CreateLitVertex((TamanyEixos) * (1 - LRdC), (GRdC),
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixXCap,
EstructuraVis.Preferencies.ColorEixXCap, 0, 0)

m_VertexsDeCapsDeEixos(28) = CreateLitVertex((TamanyEixos) * (1 - LRdC), 0, 0,
EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0, 0)
m_VertexsDeCapsDeEixos(29) = CreateLitVertex((TamanyEixos) * (1 - LRdC), -(GRdC),
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0,
0)
m_VertexsDeCapsDeEixos(30) = CreateLitVertex((TamanyEixos) * (1 - LRdC), (GRdC),
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0,
0)

m_VertexsDeCapsDeEixos(31) = CreateLitVertex((TamanyEixos) * (1 - LRdC), 0, 0,
EstructuraVis.Preferencies.ColorEixXCap, EstructuraVis.Preferencies.ColorEixXCap, 0, 0)
m_VertexsDeCapsDeEixos(32) = CreateLitVertex((TamanyEixos) * (1 - LRdC), -(GRdC),
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixXCap,
EstructuraVis.Preferencies.ColorEixXCap, 0, 0)
m_VertexsDeCapsDeEixos(33) = CreateLitVertex((TamanyEixos) * (1 - LRdC), -(GRdC) *
3, (GRdC), EstructuraVis.Preferencies.ColorEixXCap,
EstructuraVis.Preferencies.ColorEixXCap, 0, 0)
```

```
m_VertexsDeCapsDeEixos(34) = CreateLitVertex((TamanyEixos) * (1 - LRdC), 0, 0,
EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0, 0)
m_VertexsDeCapsDeEixos(35) = CreateLitVertex((TamanyEixos) * (1 - LRdC), -(GRdC) *
3, -(GRdC), EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0,
0)
m_VertexsDeCapsDeEixos(36) = CreateLitVertex((TamanyEixos) * (1 - LRdC), -(GRdC) *
3, (GRdC), EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0,
0)

m_VertexsDeCapsDeEixos(37) = CreateLitVertex((TamanyEixos) * (1 - LRdC), 0, 0,
EstructuraVis.Preferencies.ColorEixXCap, EstructuraVis.Preferencies.ColorEixXCap, 0, 0)
m_VertexsDeCapsDeEixos(38) = CreateLitVertex((TamanyEixos) * (1 - LRdC), -(GRdC) *
3, -(GRdC), EstructuraVis.Preferencies.ColorEixXCap,
EstructuraVis.Preferencies.ColorEixXCap, 0, 0)
m_VertexsDeCapsDeEixos(39) = CreateLitVertex((TamanyEixos) * (1 - LRdC), -(GRdC), -
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixXCap,
EstructuraVis.Preferencies.ColorEixXCap, 0, 0)

m_VertexsDeCapsDeEixos(40) = CreateLitVertex((TamanyEixos) * (1 - LRdC), 0, 0,
EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0, 0)
m_VertexsDeCapsDeEixos(41) = CreateLitVertex((TamanyEixos) * (1 - LRdC), (GRdC), -
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0,
0)
m_VertexsDeCapsDeEixos(42) = CreateLitVertex((TamanyEixos) * (1 - LRdC), -(GRdC), -
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixX, EstructuraVis.Preferencies.ColorEixX, 0,
0)

m_VertexsDeCapsDeEixos(43) = CreateLitVertex((TamanyEixos) * (1 - LRdC), 0, 0,
EstructuraVis.Preferencies.ColorEixXCap, EstructuraVis.Preferencies.ColorEixXCap, 0, 0)
m_VertexsDeCapsDeEixos(44) = CreateLitVertex((TamanyEixos) * (1 - LRdC), (GRdC) * 3,
-(GRdC), EstructuraVis.Preferencies.ColorEixXCap,
EstructuraVis.Preferencies.ColorEixXCap, 0, 0)
m_VertexsDeCapsDeEixos(45) = CreateLitVertex((TamanyEixos) * (1 - LRdC), (GRdC), -
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixXCap,
EstructuraVis.Preferencies.ColorEixXCap, 0, 0)



'Cap Y
m_VertexsDeCapsDeEixos(49) = CreateLitVertex(0, (TamanyEixos), 0,
EstructuraVis.Preferencies.ColorEixYCap, EstructuraVis.Preferencies.ColorEixYCap, 0, 0)
m_VertexsDeCapsDeEixos(50) = CreateLitVertex((GRdC) * 3, (TamanyEixos) * (1 - LRdC),
(GRdC), EstructuraVis.Preferencies.ColorEixYCap,
EstructuraVis.Preferencies.ColorEixYCap, 0, 0)
m_VertexsDeCapsDeEixos(51) = CreateLitVertex((GRdC) * 3, (TamanyEixos) * (1 - LRdC),
-(GRdC), EstructuraVis.Preferencies.ColorEixYCap,
EstructuraVis.Preferencies.ColorEixYCap, 0, 0)

m_VertexsDeCapsDeEixos(52) = CreateLitVertex(0, (TamanyEixos), 0,
EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0, 0)
```

```
m_VertexsDeCapsDeEixos(53) = CreateLitVertex((GRdC) * 3, (TamanyEixos) * (1 - LRdC),
(GRdC), EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0, 0)
m_VertexsDeCapsDeEixos(54) = CreateLitVertex((GRdC), (TamanyEixos) * (1 - LRdC),
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0,
0)

m_VertexsDeCapsDeEixos(55) = CreateLitVertex(0, (TamanyEixos), 0,
EstructuraVis.Preferencies.ColorEixYCap, EstructuraVis.Preferencies.ColorEixYCap, 0, 0)
m_VertexsDeCapsDeEixos(56) = CreateLitVertex(-(GRdC), (TamanyEixos) * (1 - LRdC),
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixYCap,
EstructuraVis.Preferencies.ColorEixYCap, 0, 0)
m_VertexsDeCapsDeEixos(57) = CreateLitVertex((GRdC), (TamanyEixos) * (1 - LRdC),
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixYCap,
EstructuraVis.Preferencies.ColorEixYCap, 0, 0)

m_VertexsDeCapsDeEixos(58) = CreateLitVertex(0, (TamanyEixos), 0,
EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0, 0)
m_VertexsDeCapsDeEixos(59) = CreateLitVertex(-(GRdC), (TamanyEixos) * (1 - LRdC),
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0,
0)
m_VertexsDeCapsDeEixos(60) = CreateLitVertex(-(GRdC) * 3, (TamanyEixos) * (1 -
LRdC), (GRdC), EstructuraVis.Preferencies.ColorEixY,
EstructuraVis.Preferencies.ColorEixY, 0, 0)

m_VertexsDeCapsDeEixos(61) = CreateLitVertex(0, (TamanyEixos), 0,
EstructuraVis.Preferencies.ColorEixYCap, EstructuraVis.Preferencies.ColorEixYCap, 0, 0)
m_VertexsDeCapsDeEixos(62) = CreateLitVertex(-(GRdC) * 3, (TamanyEixos) * (1 -
LRdC), -(GRdC), EstructuraVis.Preferencies.ColorEixYCap,
EstructuraVis.Preferencies.ColorEixYCap, 0, 0)
m_VertexsDeCapsDeEixos(63) = CreateLitVertex(-(GRdC) * 3, (TamanyEixos) * (1 -
LRdC), (GRdC), EstructuraVis.Preferencies.ColorEixYCap,
EstructuraVis.Preferencies.ColorEixYCap, 0, 0)

m_VertexsDeCapsDeEixos(64) = CreateLitVertex(0, (TamanyEixos), 0,
EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0, 0)
m_VertexsDeCapsDeEixos(65) = CreateLitVertex(-(GRdC) * 3, (TamanyEixos) * (1 -
LRdC), -(GRdC), EstructuraVis.Preferencies.ColorEixY,
EstructuraVis.Preferencies.ColorEixY, 0, 0)
m_VertexsDeCapsDeEixos(66) = CreateLitVertex(-(GRdC), (TamanyEixos) * (1 - LRdC), -
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0,
0)

m_VertexsDeCapsDeEixos(67) = CreateLitVertex(0, (TamanyEixos), 0,
EstructuraVis.Preferencies.ColorEixYCap, EstructuraVis.Preferencies.ColorEixYCap, 0, 0)
m_VertexsDeCapsDeEixos(68) = CreateLitVertex((GRdC), (TamanyEixos) * (1 - LRdC), -
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixYCap,
EstructuraVis.Preferencies.ColorEixYCap, 0, 0)
m_VertexsDeCapsDeEixos(69) = CreateLitVertex(-(GRdC), (TamanyEixos) * (1 - LRdC), -
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixYCap,
EstructuraVis.Preferencies.ColorEixYCap, 0, 0)
```

```
m_VertexsDeCapsDeEixos(70) = CreateLitVertex(0, (TamanyEixos), 0,
EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0, 0)
m_VertexsDeCapsDeEixos(71) = CreateLitVertex((GRdC) * 3, (TamanyEixos) * (1 - LRdC),
-(GRdC), EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0, 0)
m_VertexsDeCapsDeEixos(72) = CreateLitVertex((GRdC), (TamanyEixos) * (1 - LRdC), -
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0,
0)


m_VertexsDeCapsDeEixos(73) = CreateLitVertex(0, (TamanyEixos) * (1 - LRdC), 0,
EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0, 0)
m_VertexsDeCapsDeEixos(74) = CreateLitVertex((GRdC) * 3, (TamanyEixos) * (1 - LRdC),
(GRdC), EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0, 0)
m_VertexsDeCapsDeEixos(75) = CreateLitVertex((GRdC) * 3, (TamanyEixos) * (1 - LRdC),
-(GRdC), EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0, 0)

m_VertexsDeCapsDeEixos(76) = CreateLitVertex(0, (TamanyEixos) * (1 - LRdC), 0,
EstructuraVis.Preferencies.ColorEixYCap, EstructuraVis.Preferencies.ColorEixYCap, 0, 0)
m_VertexsDeCapsDeEixos(77) = CreateLitVertex((GRdC) * 3, (TamanyEixos) * (1 - LRdC),
(GRdC), EstructuraVis.Preferencies.ColorEixYCap,
EstructuraVis.Preferencies.ColorEixYCap, 0, 0)
m_VertexsDeCapsDeEixos(78) = CreateLitVertex((GRdC), (TamanyEixos) * (1 - LRdC),
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixYCap,
EstructuraVis.Preferencies.ColorEixYCap, 0, 0)

m_VertexsDeCapsDeEixos(79) = CreateLitVertex(0, (TamanyEixos) * (1 - LRdC), 0,
EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0, 0)
m_VertexsDeCapsDeEixos(80) = CreateLitVertex(-(GRdC), (TamanyEixos) * (1 - LRdC),
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0,
0)
m_VertexsDeCapsDeEixos(81) = CreateLitVertex((GRdC), (TamanyEixos) * (1 - LRdC),
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0,
0)

m_VertexsDeCapsDeEixos(82) = CreateLitVertex(0, (TamanyEixos) * (1 - LRdC), 0,
EstructuraVis.Preferencies.ColorEixYCap, EstructuraVis.Preferencies.ColorEixYCap, 0, 0)
m_VertexsDeCapsDeEixos(83) = CreateLitVertex(-(GRdC), (TamanyEixos) * (1 - LRdC),
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixYCap,
EstructuraVis.Preferencies.ColorEixYCap, 0, 0)
m_VertexsDeCapsDeEixos(84) = CreateLitVertex(-(GRdC) * 3, (TamanyEixos) * (1 -
LRdC), (GRdC), EstructuraVis.Preferencies.ColorEixYCap,
EstructuraVis.Preferencies.ColorEixYCap, 0, 0)

m_VertexsDeCapsDeEixos(85) = CreateLitVertex(0, (TamanyEixos) * (1 - LRdC), 0,
EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0, 0)
m_VertexsDeCapsDeEixos(86) = CreateLitVertex(-(GRdC) * 3, (TamanyEixos) * (1 -
LRdC), -(GRdC), EstructuraVis.Preferencies.ColorEixY,
EstructuraVis.Preferencies.ColorEixY, 0, 0)
```

```
m_VertexsDeCapsDeEixos(87) = CreateLitVertex(-(GRdC) * 3, (TamanyEixos) * (1 -
LRdC), (GRdC), EstructuraVis.Preferencies.ColorEixY,
EstructuraVis.Preferencies.ColorEixY, 0, 0)

m_VertexsDeCapsDeEixos(88) = CreateLitVertex(0, (TamanyEixos) * (1 - LRdC), 0,
EstructuraVis.Preferencies.ColorEixYCap, EstructuraVis.Preferencies.ColorEixYCap, 0, 0)
m_VertexsDeCapsDeEixos(89) = CreateLitVertex(-(GRdC) * 3, (TamanyEixos) * (1 -
LRdC), -(GRdC), EstructuraVis.Preferencies.ColorEixYCap,
EstructuraVis.Preferencies.ColorEixYCap, 0, 0)
m_VertexsDeCapsDeEixos(90) = CreateLitVertex(-(GRdC), (TamanyEixos) * (1 - LRdC), -
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixYCap,
EstructuraVis.Preferencies.ColorEixYCap, 0, 0)

m_VertexsDeCapsDeEixos(91) = CreateLitVertex(0, (TamanyEixos) * (1 - LRdC), 0,
EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0, 0)
m_VertexsDeCapsDeEixos(92) = CreateLitVertex((GRdC), (TamanyEixos) * (1 - LRdC), -
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0,
0)
m_VertexsDeCapsDeEixos(93) = CreateLitVertex(-(GRdC), (TamanyEixos) * (1 - LRdC), -
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixY, EstructuraVis.Preferencies.ColorEixY, 0,
0)

m_VertexsDeCapsDeEixos(94) = CreateLitVertex(0, (TamanyEixos) * (1 - LRdC), 0,
EstructuraVis.Preferencies.ColorEixYCap, EstructuraVis.Preferencies.ColorEixYCap, 0, 0)
m_VertexsDeCapsDeEixos(95) = CreateLitVertex((GRdC) * 3, (TamanyEixos) * (1 - LRdC),
-(GRdC), EstructuraVis.Preferencies.ColorEixYCap,
EstructuraVis.Preferencies.ColorEixYCap, 0, 0)
m_VertexsDeCapsDeEixos(96) = CreateLitVertex((GRdC), (TamanyEixos) * (1 - LRdC), -
(GRdC) * 3, EstructuraVis.Preferencies.ColorEixYCap,
EstructuraVis.Preferencies.ColorEixYCap, 0, 0)



'Cap Z
m_VertexsDeCapsDeEixos(97) = CreateLitVertex(0, 0, (TamanyEixos),
EstructuraVis.Preferencies.ColorEixZCap, EstructuraVis.Preferencies.ColorEixZCap, 0, 0)
m_VertexsDeCapsDeEixos(98) = CreateLitVertex((GRdC) * 3, (GRdC), (TamanyEixos) * (1
- LRdC), EstructuraVis.Preferencies.ColorEixZCap,
EstructuraVis.Preferencies.ColorEixZCap, 0, 0)
m_VertexsDeCapsDeEixos(99) = CreateLitVertex((GRdC) * 3, -(GRdC), (TamanyEixos) * (1
- LRdC), EstructuraVis.Preferencies.ColorEixZCap,
EstructuraVis.Preferencies.ColorEixZCap, 0, 0)

m_VertexsDeCapsDeEixos(100) = CreateLitVertex(0, 0, (TamanyEixos),
EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0, 0)
m_VertexsDeCapsDeEixos(101) = CreateLitVertex((GRdC) * 3, (GRdC), (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0,
0)
m_VertexsDeCapsDeEixos(102) = CreateLitVertex((GRdC), (GRdC) * 3, (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0,
0)
```

```
m_VertexsDeCapsDeEixos(103) = CreateLitVertex(0, 0, (TamanyEixos),
EstructuraVis.Preferencies.ColorEixZCap, EstructuraVis.Preferencies.ColorEixZCap, 0, 0)
m_VertexsDeCapsDeEixos(104) = CreateLitVertex(-(GRdC), (GRdC) * 3, (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZCap,
EstructuraVis.Preferencies.ColorEixZCap, 0, 0)
m_VertexsDeCapsDeEixos(105) = CreateLitVertex((GRdC), (GRdC) * 3, (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZCap,
EstructuraVis.Preferencies.ColorEixZCap, 0, 0)

m_VertexsDeCapsDeEixos(106) = CreateLitVertex(0, 0, (TamanyEixos),
EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0, 0)
m_VertexsDeCapsDeEixos(107) = CreateLitVertex(-(GRdC), (GRdC) * 3, (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0,
0)
m_VertexsDeCapsDeEixos(108) = CreateLitVertex(-(GRdC) * 3, (GRdC), (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0,
0)

m_VertexsDeCapsDeEixos(109) = CreateLitVertex(0, 0, (TamanyEixos),
EstructuraVis.Preferencies.ColorEixZCap, EstructuraVis.Preferencies.ColorEixZCap, 0, 0)
m_VertexsDeCapsDeEixos(110) = CreateLitVertex(-(GRdC) * 3, -(GRdC), (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZCap,
EstructuraVis.Preferencies.ColorEixZCap, 0, 0)
m_VertexsDeCapsDeEixos(111) = CreateLitVertex(-(GRdC) * 3, (GRdC), (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZCap,
EstructuraVis.Preferencies.ColorEixZCap, 0, 0)

m_VertexsDeCapsDeEixos(112) = CreateLitVertex(0, 0, (TamanyEixos),
EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0, 0)
m_VertexsDeCapsDeEixos(113) = CreateLitVertex(-(GRdC) * 3, -(GRdC), (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0,
0)
m_VertexsDeCapsDeEixos(114) = CreateLitVertex(-(GRdC), -(GRdC) * 3, (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0,
0)

m_VertexsDeCapsDeEixos(115) = CreateLitVertex(0, 0, (TamanyEixos),
EstructuraVis.Preferencies.ColorEixZCap, EstructuraVis.Preferencies.ColorEixZCap, 0, 0)
m_VertexsDeCapsDeEixos(116) = CreateLitVertex((GRdC), -(GRdC) * 3, (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZCap,
EstructuraVis.Preferencies.ColorEixZCap, 0, 0)
m_VertexsDeCapsDeEixos(117) = CreateLitVertex(-(GRdC), -(GRdC) * 3, (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZCap,
EstructuraVis.Preferencies.ColorEixZCap, 0, 0)

m_VertexsDeCapsDeEixos(118) = CreateLitVertex(0, 0, (TamanyEixos),
EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0, 0)
m_VertexsDeCapsDeEixos(119) = CreateLitVertex((GRdC) * 3, -(GRdC), (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0,
0)
```

```
m_VertexsDeCapsDeEixos(120) = CreateLitVertex((GRdC), -(GRdC) * 3, (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0,
0)



m_VertexsDeCapsDeEixos(121) = CreateLitVertex(0, 0, (TamanyEixos) * (1 - LRdC),
EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0, 0)
m_VertexsDeCapsDeEixos(122) = CreateLitVertex((GRdC) * 3, (GRdC), (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0,
0)
m_VertexsDeCapsDeEixos(123) = CreateLitVertex((GRdC) * 3, -(GRdC), (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0,
0)

m_VertexsDeCapsDeEixos(124) = CreateLitVertex(0, 0, (TamanyEixos) * (1 - LRdC),
EstructuraVis.Preferencies.ColorEixZCap, EstructuraVis.Preferencies.ColorEixZCap, 0, 0)
m_VertexsDeCapsDeEixos(125) = CreateLitVertex((GRdC) * 3, (GRdC), (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZCap,
EstructuraVis.Preferencies.ColorEixZCap, 0, 0)
m_VertexsDeCapsDeEixos(126) = CreateLitVertex((GRdC), (GRdC) * 3, (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZCap,
EstructuraVis.Preferencies.ColorEixZCap, 0, 0)

m_VertexsDeCapsDeEixos(127) = CreateLitVertex(0, 0, (TamanyEixos) * (1 - LRdC),
EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0, 0)
m_VertexsDeCapsDeEixos(128) = CreateLitVertex(-(GRdC), (GRdC) * 3, (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0,
0)
m_VertexsDeCapsDeEixos(129) = CreateLitVertex((GRdC), (GRdC) * 3, (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0,
0)

m_VertexsDeCapsDeEixos(130) = CreateLitVertex(0, 0, (TamanyEixos) * (1 - LRdC),
EstructuraVis.Preferencies.ColorEixZCap, EstructuraVis.Preferencies.ColorEixZCap, 0, 0)
m_VertexsDeCapsDeEixos(131) = CreateLitVertex(-(GRdC), (GRdC) * 3, (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZCap,
EstructuraVis.Preferencies.ColorEixZCap, 0, 0)
m_VertexsDeCapsDeEixos(132) = CreateLitVertex(-(GRdC) * 3, (GRdC), (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZCap,
EstructuraVis.Preferencies.ColorEixZCap, 0, 0)

m_VertexsDeCapsDeEixos(133) = CreateLitVertex(0, 0, (TamanyEixos) * (1 - LRdC),
EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0, 0)
m_VertexsDeCapsDeEixos(134) = CreateLitVertex(-(GRdC) * 3, -(GRdC), (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0,
0)
m_VertexsDeCapsDeEixos(135) = CreateLitVertex(-(GRdC) * 3, (GRdC), (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0,
0)
```

```
m_VertexsDeCapsDeEixos(136) = CreateLitVertex(0, 0, (TamanyEixos) * (1 - LRdC),
EstructuraVis.Preferencies.ColorEixZCap, EstructuraVis.Preferencies.ColorEixZCap, 0, 0)
m_VertexsDeCapsDeEixos(137) = CreateLitVertex(-(GRdC) * 3, -(GRdC), (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZCap,
EstructuraVis.Preferencies.ColorEixZCap, 0, 0)
m_VertexsDeCapsDeEixos(138) = CreateLitVertex(-(GRdC), -(GRdC) * 3, (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZCap,
EstructuraVis.Preferencies.ColorEixZCap, 0, 0)

m_VertexsDeCapsDeEixos(139) = CreateLitVertex(0, 0, (TamanyEixos) * (1 - LRdC),
EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0, 0)
m_VertexsDeCapsDeEixos(140) = CreateLitVertex((GRdC), -(GRdC) * 3, (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0,
0)
m_VertexsDeCapsDeEixos(141) = CreateLitVertex(-(GRdC), -(GRdC) * 3, (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZ, EstructuraVis.Preferencies.ColorEixZ, 0,
0)

m_VertexsDeCapsDeEixos(142) = CreateLitVertex(0, 0, (TamanyEixos) * (1 - LRdC),
EstructuraVis.Preferencies.ColorEixZCap, EstructuraVis.Preferencies.ColorEixZCap, 0, 0)
m_VertexsDeCapsDeEixos(143) = CreateLitVertex((GRdC) * 3, -(GRdC), (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZCap,
EstructuraVis.Preferencies.ColorEixZCap, 0, 0)
m_VertexsDeCapsDeEixos(144) = CreateLitVertex((GRdC), -(GRdC) * 3, (TamanyEixos) *
(1 - LRdC), EstructuraVis.Preferencies.ColorEixZCap,
EstructuraVis.Preferencies.ColorEixZCap, 0, 0)


Set EstructuraVis.DadesDirectX.VBufferDeCapsDeEixosV3 =
EstructuraVis.DadesDirectX.D3DDeviceV3.CreateVertexBuffer _
 (Len(m_VertexsDeCapsDeEixos(1)) * 144, 0, TL_FVF, D3DPOOL_DEFAULT)
'Set EstructuraVis.DadesDirectX.VBufferDeCapsDeEixosV2 =
EstructuraVis.DadesDirectX.D3DDeviceV2.CreateVertexBuffer _
 (Len(m_VertexsDeCapsDeEixos(1)) * 144, 0, TL_FVF, D3DPOOL_DEFAULT)
'Set EstructuraVis.DadesDirectX.VBufferDeCapsDeEixosV1 =
EstructuraVis.DadesDirectX.D3DDeviceV1.CreateVertexBuffer _
 (Len(m_VertexsDeCapsDeEixos(1)) * 144, 0, TL_FVF, D3DPOOL_DEFAULT)
'Set EstructuraVis.DadesDirectX.VBufferDeCapsDeEixosV4 =
EstructuraVis.DadesDirectX.D3DDeviceV4.CreateVertexBuffer _
 (Len(m_VertexsDeCapsDeEixos(1)) * 144, 0, TL_FVF, D3DPOOL_DEFAULT)

'I fill the created vertex with the data

        D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferDeCapsDeEixosV3,
0, _
        Len(m_VertexsDeCapsDeEixos(1)) * 144, 0, m_VertexsDeCapsDeEixos(1)
        'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferDeCapsDeEixosV2,
0, _
        Len(m_VertexsDeCapsDeEixos(1)) * 144, 0, m_VertexsDeCapsDeEixos(1)
        'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferDeCapsDeEixosV1,
0, _
```

```
        Len(m_VertexsDeCapsDeEixos(1)) * 144, 0, m_VertexsDeCapsDeEixos(1)
        'D3DVertexBuffer8SetData EstructuraVis.DadesDirectX.VBufferDeCapsDeEixosV4,
0, _
        Len(m_VertexsDeCapsDeEixos(1)) * 144, 0, m_VertexsDeCapsDeEixos(1)

End Sub
```