

DCOM Wrapper Examples of Various Data Types

This chapter provides Visual Basic examples of how programs generated with the IDL generator are used with various data types.

Data Type A<nn>

IDL

```
Program 'ConstA' Is
  Define Data Parameter
    1 Client      (A80) In
    1 Mail        (A80) In Out
  End-Define
```

Visual Basic Code

```
Function GetMail(S1 As String, S2 As String) As String
  Dim Client As String * 80
  Dim Mail As String * 80

  Client = S1
  Mail = S2

  obj.ConstA Client, Mail ' Perform call to dcom object

  GetMail = Mail
End Function
```

Data Type A<nn>/<nn>

IDL

```
Program 'ConstAN' Is
  Define Data Parameter
    1 Client      (A80/3) In
    1 Mail        (A80/2) In Out
  End-Define
```

Visual Basic Code

```
Private Function GetMail(S1 As String, S2 As String) As String()
  Dim Client(2) As String
  Dim Mail(1) As String
  Dim tempStr As String * 80 ' string with fixed size

  For i = 0 To UBound(Client)
    tempStr = S1
    Client(i) = tempStr
  Next

  For i = 0 To UBound(Mail)
    tempStr = S2
```

```

        Mail(i) = tempStr
    Next

    obj.ConstAN Client, Mail    ' Perform call to dcom object

    GetMail = Mail
End Function

```

Data Type AV

IDL

```

Program 'ArrayAV' Is
    Define Data Parameter
        1 Client    (AV)  In
        1 Mail      (AV)  In Out
    End-Define

```

Visual Basic Code

```

Private Function GetMail(S1 As String, S2 As String) As String()
    Dim Client As String
    Dim Mail As String

    Client = S1
    Mail = S2

    obj.ArrayAV Client, Mail    ' Perform call to dcom object

    GetMail = Mail
End Function

```

Data Type AV/V,V

IDL

```

Program 'ArrayAVVV' Is
    Define Data Parameter
        1 Client    (AV/V,V)  In
        1 Mail      (AV/V,V)  In Out
    End-Define

```

Visual Basic Code

```

Private Function GetMail(S1 As String, S2 As String) As Variant()
    Dim Client() As Variant    ' ** define array parameter
    Dim Mail() As Variant     ' ** define array parameter

    ReDim Client(4)
    ReDim Mail(2)

    Dim m_str() As String

    For i = 0 To UBound(Client)
        ReDim m_str(2)
        For j = 0 To UBound(m_str)
            m_str(j) = Str(i) & Str(j) & S1 ' assign a string
        Next j
    Next i

```

```

        Client(i) = m_str ' assign a new array
    Next i

    For i = 0 To UBound(Mail)
        ReDim m_str(1)
        For j = 0 To UBound(m_str)
            m_str(j) = Str(i) & Str(j) & S2 ' assign a string
        Next j
        Mail(i) = m_str ' assign a new array
    Next i

    obj.ArrayAVVV Client, Mail ' Perform call to dcom object

    GetMail = Mail
End Function

```

Data Type AV/<nn>

IDL

```

Program 'ArrayAVnn' Is
    Define Data Parameter
        1 Client      (AV/2)  In
        1 Mail        (AV/5)  In Out
        1 Server      (AV/7)  Out
    End-Define

```

Visual Basic Code

```

Private Function GetMail(S1 As String, S2 As String) As String()
    Dim Client() As String
    Dim Mail() As String
    Dim Server() As String

    ReDim Client(1) '2 elements
    ReDim Mail(4) '5 elements

    For i = 0 To UBound(Client)
        Client(i) = S1 & " " & CStr(i)
    Next i

    For i = 0 To UBound(Mail)
        Mail(i) = S2 & " " & CStr(i)
    Next i

    obj.ArrayAVnn Client, Mail, Server ' Perform call to dcom object

    GetMail = Server
End Function

```

Data Type AV/<nn>,<nn>

IDL

```

Program 'ArrayAVnnnn' Is
    Define Data Parameter

```

```

1 Client      (AV/2,3)  In
1 Mail       (AV/5,7)  In Out
1 Server     (AV/7,9)  Out
End-Define

```

Visual Basic Code

```

Private Function GetMail(S1 As String, S2 As String) As Variant()
    Dim Client() As String
    Dim Mail()   As String
    Dim Server() As String

    ReDim Client(1, 2)      '2,3 elements
    ReDim Mail(4, 6)       '5,7 elements

    For i = 0 To UBound(Client, 1)
        For j = 0 To UBound(Client, 2)
            Client(i, j) = S1 & " " & CStr(i) & " " & CStr(j)
        Next j
    Next i

    For i = 0 To UBound(Mail, 1)
        For j = 0 To UBound(Mail, 2)
            Mail(i, j) = S2 & " " & CStr(i) & " " & CStr(j)
        Next j
    Next i

    obj.ArrayAVnnnn Client, Mail, Server ' Perform call to dcom object

    GetMail = Server
End Function

```

Data Type AV/<nn>,<nn>,<nn>

IDL

```

Program 'ArrayAVnnnnnn' Is
    Define Data Parameter
        1 Client      (AV/2,3,4)  In
        1 Mail       (AV/5,7,8)  In Out
        1 Server     (AV/9,6,3)  Out
    End-Define

```

See illustration for client (AV/2,3,4) under *Example of Arrays with Fixed Bounds*.

Visual Basic Code

```

Private Function GetMail(S1 As String, S2 As String) As Variant()
    Dim Client() As String
    Dim Mail()   As String
    Dim Server() As String

    ReDim Client(1, 2, 3)      '2,3,4 elements
    ReDim Mail(4, 6, 7)       '5,7,8 elements

    For i = 0 To UBound(Client, 1)
        For j = 0 To UBound(Client, 2)
            For k = 0 To UBound(Client, 3)
                Client(i, j, k) = S1 & " " & CStr(i) & " " & CStr(j) & " "
                & CStr(k)
            Next k
        Next j
    Next i

```

```

        Next k
    Next j
Next i

For i = 0 To UBound(Mail, 1)
    For j = 0 To UBound(Mail, 2)
        For k = 0 To UBound(Mail, 3)
            Mail(i, j, k) = S2 & " " & CStr(i) & " " & CStr(j) & " "
& CStr(k)
        Next k
    Next j
Next i

obj.ArrayAVnnnnnn Client, Mail, Server ' Perform call to dcom object

GetMail = Server
End Function

```

Data Type AV/V,V,V

IDL

```

Program 'ArrayAVVVV' Is
    Define Data Parameter
        1 Client      (AV/V,V,V)  In
        1 Mail        (AV/V,V,V)  In Out
        1 Server      (AV/V,V,V)  In Out
    End-Define

```

See illustration for client (AV/V,V,V) under *Three-dimensional Array with Variable Upper Bounds*.

Visual Basic Code

```

Private Function GetMail(S1 As String, S2 As String) As Variant()
    Dim Client() As Variant
    Dim Mail()   As Variant
    Dim Server() As Variant

    ReDim Client(4)
    ReDim Mail(2)

    Dim m_str() As String
    Dim m_Array() As Variant

    For i = 0 To UBound(Client)
        ReDim m_Array(8)
        For j = 0 To UBound(m_Array)
            ReDim m_str(2)
            For k = 0 To UBound(m_str)
                m_str(k) = S1 & Str(i) & Str(j) & S1 ' assign a string
            Next k
            m_Array(j) = m_str
        Next j
        Client(i) = m_str ' assign a new array
    Next i

    For i = 0 To UBound(Mail)
        ReDim m_Array(8)
        For j = 0 To UBound(m_Array)
            ReDim m_str(2)

```

```

    For k = 0 To UBound(m_str)
        m_str(k) = S1 & Str(i) & Str(j) & S2 ' assign a string
    Next k
    m_Array(j) = m_str
    Next j
    Mail(i) = m_str ' assign a new array
Next i

obj.ArrayAVVV Client, Mail, Server ' Perform call to dcom object

GetMail = Mail
End Function

```

Data Type I<nn>/<nn>

IDL

```

Program 'ConstInn' Is
  Define Data Parameter
    1 Client      (I4/4)  In
    1 Mail        (I4/5)  In Out
  End-Define

```

Visual Basic Code

```

Private Function GetMail(L1 As Long, L2 As Long) As Long()
  Dim Client(3) As Long ' define I4 as Long, I2 as Integer and I1 as Integer
  Dim Mail(4) As Long

  For i = 0 To UBound(Client)
    Client(i) = L1
  Next

  For i = 0 To UBound(Mail)
    Mail(i) = L2
  Next

  obj.ConstInn Client, Mail ' Perform call to dcom object

  GetMail = Mail
End Function

```

Data Type I2/V,V

IDL

```

Program 'Int2VV' Is
  Define Data Parameter
    1 Client      (I2/V,V)  In
    1 Mail        (I2/V,V)  In Out
  End-Define

```

Visual Basic Code

```

Private Function GetMail(I1 As Integer, I2 As Integer) As Integer()
    Dim Client() As Variant ' ** define array parameter
    Dim Mail() As Variant ' ** define array parameter

    Dim ClientValue() As Integer
    Dim MailValue() As Integer

    ReDim Client(8)
    ReDim Mail(3)
    ReDim ClientValue(4)
    ReDim MailValue(5)

    For i = 0 To UBound(Client)
        For j = 0 To UBound(ClientValue)
            ClientValue(j) = i * j + I1 ' write any data into the elements
        Next j
        Client(i) = ClientValue ' assign the new array to an element of the main array
    Next i

    For i = 0 To UBound(Mail)
        For j = 0 To UBound(MailValue)
            MailValue(j) = i * j + I2 ' write any data into the elements
        Next j
        Mail(i) = MailValue ' assign the new array to an element of the main array
    Next i

    obj.Int2VV Client, Mail ' Perform call to dcom object

    GetMail = Mail
End Function

```

Data Type I<nn>/<nn>,<nn>

IDL

```

Program 'Int4NN' Is
    Define Data Parameter
        1 Client (I4/5,4) In
        1 Mail (I4/6,3) In Out
        1 Server (I4/3,2) Out
    End-Define

```

Visual Basic Code

```

Private Function GetMail(I1 As Long, I2 As Long) As Long()
    Dim Client() As Long ' define I4 as Long, I2 as Integer and I1 as Integer
    Dim Mail() As Long
    Dim Server() As Long

    ReDim Client(4,3)
    ReDim Mail(5,2)

    For i = 0 To UBound(Client, 1)
        For j = 0 To UBound(Client, 2)
            Client(i, j) = i * j + I1 ' write any data into the elements
        Next j
    Next i

    For i = 0 To UBound(Mail, 1)

```

```

    For j = 0 To UBound(Mail, 2)
        Mail(i, j) = i * j + I2 ' write any data into the elements
    Next j
Next i

obj.Int4VV Client, Mail, Server ' Perform call to dcom object

GetMail = Server
End Function

```

Data Type I<nn>/V,V,V

IDL

```

Program 'Int2VVV' Is
  Define Data Parameter
    1 Client      (I2/V,V,V)  In
    1 Mail        (I2/V,V,V)  In Out
    1 Server      (I2/V,V,V)  Out
  End-Define

```

Visual Basic Code

```

Private Function GetMail(I1 As Integer, I2 As Integer) As Integer()
  Dim Client() As Variant
  Dim Mail() As Variant
  Dim Server() As Variant

  Dim ClientWrap() As Variant
  Dim MailWrap() As Variant

  Dim ClientValue() As Integer ' define I4 as Long, I2 as Integer and I1 as Integer
  Dim MailValue() As Integer

  ReDim Client(4)
  ReDim Mail(5)

  ReDim ClientWrap(10)
  ReDim MailWrap(23)

  ReDim ClientValue(200)
  ReDim MailValue(238)

  For i = 0 To UBound(Client)
    For j = 0 To UBound(ClientWrap)
      For k = 0 To UBound(ClientValue)
        ClientValue(k) = k + i * j + I1 ' write any data into the elements
      Next k
      ClientWrap(j) = ClientValue
    Next j
    Client(i) = ClientWrap
  Next i

  For i = 0 To UBound(Mail)
    For j = 0 To UBound(MailWrap)
      For k = 0 To UBound(MailValue)
        MailValue(k) = k + i * j + I2 ' write any data into the elements
      Next k
      MailWrap(j) = MailValue
    Next j
  Next i

```



```

    Mail(i) = MailWrap
Next i

obj.Int2VWV Client, Mail, Server    ' Perform call to dcom object

GetMail = Server
End Function

```

Data Type NU<nn>/V,V

IDL

```

Program 'NUnV' Is
  Define Data Parameter
    1 Client    (NU21/V,V)  In
    1 Mail      (NU21/V,V)  In Out
  End-Define

```

Visual Basic Code

```

Private Function GetMail() As String()
  Dim Client() As Variant    ' ** define array parameter
  Dim Mail()   As Variant    ' ** define array parameter

  Dim ClientValue() As String
  Dim MailValue()   As String

  ReDim Client(8)
  ReDim Mail(3)
  ReDim ClientValue(4)
  ReDim MailValue(5)

  For i = 0 To UBound(Client)
    For j = 0 To UBound(ClientValue)
      ClientValue(j) = "55456698" ' write any data into the elements
    Next j
    Client(i) = ClientValue      ' assign the new array to an element of the main array
  Next i

  For i = 0 To UBound(Mail)
    For j = 0 To UBound(MailValue)
      MailValue(j) = "548799875465" ' write any data into the elements
    Next j
    Mail(i) = MailValue          ' assign the new array to an element of the main array
  Next i

  obj.NUnV Client, Mail    ' Perform call to dcom object

  GetMail = Mail
End Function

```

Data Type P<nn>.<nn>/V,V

IDL

```

Program 'PnXV' Is

```

```

Define Data Parameter
    1 Client      (P12.2/V,V)  In
    1 Mail        (P12.2/V,V)  In Out
End-Define

```

Visual Basic Code

```

Private Function GetMail() As String()
    Dim Client() As Variant ' ** define array parameter
    Dim Mail() As Variant ' ** define array parameter

    Dim ClientValue() As String
    Dim MailValue() As String

    ReDim Client(8)
    ReDim Mail(3)
    ReDim ClientValue(4)
    ReDim MailValue(5)

    For i = 0 To UBound(Client)
        For j = 0 To UBound(ClientValue)
            ClientValue(j) = "1234568" & "." & "65" ' write any data into the
elements
        Next j
        Client(i) = ClientValue ' assign the new array to an element of the main array
    Next i

    For i = 0 To UBound(Mail)
        For j = 0 To UBound(MailValue)
            MailValue(j) = "8799875" & "." & "32" ' write any data into the
elements
        Next j
        Mail(i) = MailValue ' assign the new array to an element of the main array
    Next i

    obj.PnXVW Client, Mail ' Perform call to dcom object

    GetMail = Mail
End Function

```

Data Type L/V

IDL

```

Program 'LV' Is
    Define Data Parameter
        1 Client      (L/V)  In
        1 Mail        (L/V)  In Out
        1 Server      (L/V)  Out
    End-Define

```

Visual Basic Code

```

Private Function GetMail() As Boolean()
    Dim Client() As Boolean
    Dim Mail() As Boolean
    Dim Server() As Boolean

    ReDim Client(8)
    ReDim Mail(3)

```

```

For i = 0 To UBound(Client)
    Client(i) = True
Next i

For i = 0 To UBound(Mail)
    Mail(i) = False
Next i

obj.LV Client, Mail, Server    ' Perform call to dcom object

GetMail = Mail
End Function

```

Data Type D

IDL

```

Program 'DateProc' Is
    Define Data Parameter
        1 Client    (D)  In
        1 Mail      (D)  In Out
        1 Server    (D)  Out
    End-Define

```

Visual Basic Code

```

Private Function GetMail() As String
    Dim Client As Date
    Dim Mail    As Date
    Dim Server  As Date

    Dim s As String

    ' set the input data with the current date
    Client = Date
    Mail = Date

    obj.DateProc Client, Mail, Server

    ' work with the returned data
    s = Mail & " " & Server

    GetMail = s
End Function

```

Data Type D/V,V

IDL

```

Program 'DateProcVV' Is
    Define Data Parameter
        1 Client    (D/V,V)  In
        1 Mail      (D/V,V)  In Out
        1 Server    (D/V,V)  Out
    End-Define

```

Visual Basic Code

```

Private Function GetMail() As String
    Dim Client() As Variant
    Dim Mail() As Variant
    Dim Server() As Variant

    Dim Value() As Double ' Use a Double instead of a Date datatype
    Dim s As String

    Redim Client(5)
    Redim Value(12)
    For i = 0 To UBound(Client)
        For j = 0 To UBound(Value)
            ' set the input data with the current date
            Value(j) = Date
        Next j
        Client(i) = Value
    Next i

    Redim Mail(5)
    Redim Value(12)
    For i = 0 To UBound(Mail)
        For j = 0 To UBound(Value)
            ' set the input data with the current date
            Value(j) = Date
        Next j
        Mail(i) = Value
    Next i

    obj.DateProcVV Client, Mail, Server

    ' work with the returned data
    s = ""
    For i = 0 To UBound(Mail)
        Value = Mail(i)
        For j = 0 To UBound(Value)
            s = s & CDate(Value(j)) ' Convert the data to a Date data type if necessary
        Next j
        s = s & " "
    Next i

    s = ""
    For i = 0 To UBound(Server)
        Value = Server(i)
        For j = 0 To UBound(Value)
            s = s & CDate(Value(j))
        Next j
        s = s & " "
    Next i

    GetMail = s
End Function

```

Data Type F4/V

IDL

```

Program 'ProcF4V' Is
  Define Data Parameter
    1 Client      (F4/V)  In
    1 Mail        (F4/V)  In Out
    1 Server      (F4/V)  Out
  End-Define

```

Visual Basic Code

```

Private Function GetMail() As String
  Dim Client() As Single
  Dim Mail() As Single
  Dim Server() As Single

  Dim s As String

  Redim Client(5)
  For i = 0 To UBound(Client)
    Client(i) = 1.5542 + (i * 1.57)
  Next i

  Redim Mail(28)
  For i = 0 To UBound(Mail)
    Mail(i) = -1.554 + (i * 1.57)
  Next i

  obj.ProcF4V Client, Mail, Server

  s = ""
  For i = LBound(Server) To UBound(Server)
    s = " " & Str(Server(i))
  Next i

  GetMail = s
End Function

```

Data Type F8/V,V**IDL**

```

Program 'ProcF8VV' Is
  Define Data Parameter
    1 Client      (F8/V,V)  In
    1 Mail        (F8/V,V)  In Out
    1 Server      (F8/V,V)  Out
  End-Define

```

Visual Basic Code

```

Private Function GetMail() As String
  Dim Client() As Variant
  Dim Mail() As Variant
  Dim Server() As Variant

  Dim Value() As Double
  Dim s As String

```

```

Redim Client(5)
Redim Value(33)
For i = 0 To UBound(Client)
    For j = 0 To UBound(Value)
        Value(j) = 1.5587 + (j * 1.57) + i
    Next j
    Client(i) = Value
Next i

Redim Mail(42)
Redim Value(3)
For i = 0 To UBound(Mail)
    For j = 0 To UBound(Value)
        Value(j) = -5021.327 + (j * 3.889) - i
    Next j
    Mail(i) = Value
Next i

obj.ProcF8VV Client, Mail, Server

s = ""
For i = LBound(Server) To UBound(Server)
    Value = Server(i)
    For j = LBound(Value) To UBound(Value)
        s = " " & Str(Value(j))
    Next j
Next i

GetMail = s
End Function

```

Data Type B1/<nn>

IDL

```

Program 'Binlnn' Is
    Define Data Parameter
        1 Client      (B1/100)  In
        1 Mail        (B1/100)  In Out
    End-Define

```

Visual Basic Code

```

Private Function GetMail(B1 As Byte, B2 As Byte) As Byte()
    Dim Client() As Byte
    Dim Mail() As Byte

    ReDim Client(99)
    ReDim Mail(99)

    ' Client
    For j = 0 To UBound(Client)
        Client(j) = j + B1
    Next j

    'Mail
    For j = 0 To UBound(Mail)
        Mail(j) = j + B2
    Next j

```

```

obj.BinInn Client, Mail

GetMail = Mail
End Function

```

Data Type B<nn>/<nn>

IDL

```

Program 'BinXnn' Is
  Define Data Parameter
    1 Client      (B10/12) In
    1 Mail        (B10/12) In Out
  End-Define

```

Visual Basic Code

```

Private Function GetMail() As Byte()
  Dim Client() As Byte
  Dim Mail() As Byte

  ReDim Client(11, 9)
  ReDim Mail(11, 9)

  ' Client
  For j = 0 To UBound(Client, 1)
    For i = 0 To UBound(Client, 2)
      Client(j, i) = j + i
    Next i
  Next j

  'Mail
  For j = 0 To UBound(Mail, 1)
    For i = 0 To UBound(Mail, 2)
      Mail(j, i) = j + i + 6
    Next i
  Next j

  obj.BinXnn Client, Mail

  GetMail = Mail
End Function

```

Data Type BV

IDL

```

Program 'BinV' Is
  Define Data Parameter
    1 Client      (BV) In
    1 Mail        (BV) In Out
  End-Define

```

Visual Basic Code

```
Private Function GetMail() As Byte()
    Dim Client          As Variant
    Dim Mail            As Variant
    Dim ClientValue()  As Byte
    Dim MailValue()    As Byte

    ReDim ClientValue(8)
    ReDim MailValue(4)

    For j = 0 To UBound(ClientValue)
        ClientValue(j) = j + 6
    Next j

    For j = 0 To UBound(MailValue)
        MailValue(j) = j + 8
    Next j

    Client = ClientValue
    Mail = MailValue

    obj.BinV Client, Mail

    GetMail = Mail
End Function
```

Data Type BV/<nn>

IDL

```
Program 'BinVnn' Is
    Define Data Parameter
        1 Client      (BV/3)  In
        1 Mail        (BV/5)  In Out
    End-Define
```

Visual Basic Code

```
Private Function GetMail() As Byte()
    Dim Client()       As Variant
    Dim Mail()         As Variant
    Dim ClientValue() As Byte
    Dim MailValue()   As Byte

    ReDim Client(2)    ' /3
    ReDim Mail(4)      ' /4
    ReDim ClientValue(3)
    ReDim MailValue(5)

    For i = 0 To UBound(Client)
        For j = 0 To UBound(ClientValue)
            ClientValue(j) = j + 2
        Next j
        Client(i) = ClientValue
    Next i

    For i = 0 To UBound(Mail)
        For j = 0 To UBound(MailValue)
            MailValue(j) = j + 3
        Next j
    Next i
```



```

    Next j
    Mail(i) = MailValue
Next i

obj.BinVnn Client, Mail

GetMail = Mail
End Function

```

Data Type B<nn>

IDL

```

Program 'Binnn' Is
  Define Data Parameter
    1 Client      (B12) In
    1 Mail        (B30) In Out
  End-Define

```

Visual Basic Code

```

Private Function GetMail() As Variant()
  Dim Client() As Variant
  Dim Mail() As Variant

  Dim ClientValue() As Byte
  Dim MailValue() As Byte

  ReDim ClientValue(11) ' must be 11 ( = 12 elements -> B12)
  For j = 0 To UBound(ClientValue)
    ClientValue(j) = j + 6
  Next j
  Client = ClientValue

  ReDim MailValue(29) ' must be 29 ( = 30 elements -> B30)
  For j = 0 To UBound(MailValue)
    MailValue(j) = j + 6
  Next j
  Mail = MailValue

  obj.Binnn Client, Mail

  GetMail = Mail
End Function

```

Data Type BV/V

IDL

```

Program 'BinVV' Is
  Define Data Parameter
    1 Client      (BV/V) In
    1 Mail        (BV/V) In Out
  End-Define

```

Visual Basic Code

```
Private Function GetMail() As Variant()
    Dim Client() As Variant
    Dim Mail() As Variant

    Dim ClientValue() As Byte
    Dim MailValue() As Byte

    ReDim Client(5)
    ReDim ClientValue(32)
    For i = 0 To UBound(Client)
        For j = 0 To UBound(ClientValue)
            ClientValue(j) = j + i + 2
        Next j
        Client(i) = ClientValue
    Next i

    ReDim Mail(33)
    ReDim MailValue(20)
    For i = 0 To UBound(Mail)
        For j = 0 To UBound(MailValue)
            MailValue(j) = j + i + 2
        Next j
        Mail(i) = MailValue
    Next i

    obj.BinVV Client, Mail

    GetMail = Mail
End Function
```

Data Type BV<nn>

IDL

```
Program 'BinVxx' Is
    Define Data Parameter
        1 Client    (BV10)  In
        1 Mail      (BV12)  In Out
    End-Define
```

Visual Basic Code

```
Private Function GetMail() As Variant
    Dim Client As Variant
    Dim Mail As Variant

    Dim ClientValue() As Byte
    Dim MailValue() As Byte

    ReDim ClientValue(9) ' 9 ( = 10 elements) or less
    For j = 0 To UBound(ClientValue)
        ClientValue(j) = j + 6
    Next j
    Client(i) = ClientValue

    ReDim MailValue(11) ' 11 ( = 12 elements) or less
    For j = 0 To UBound(MailValue)
        MailValue(j) = j + 7
```

```

Next j
Mail(i) = MailValue

obj.BinVxx Client, Mail

GetMail = Mail
End Function

```

Data Type B<nn>/V

IDL

```

Program 'BinV' Is
  Define Data Parameter
    1 Client      (B4/V) In
    1 Mail        (B2/V) In Out
  End-Define

```

Visual Basic Code

```

Private Function GetMail() As Variant
  Dim Client()      As Variant
  Dim Mail()        As Variant
  Dim ClientValue() As Byte
  Dim MailValue()   As Byte

  ReDim Client(22)
  ReDim ClientValue(3) ' B4
  For i = 0 To UBound(Client)
    For j = 0 To UBound(ClientValue)
      ClientValue(j) = j + i + 2
    Next j
    Client(i) = ClientValue
  Next i

  ReDim Mail(15)
  ReDim MailValue(1) ' B2
  For i = 0 To UBound(Mail)
    For j = 0 To UBound(MailValue)
      MailValue(j) = j + i + 2
    Next j
    Mail(i) = MailValue
  Next i

  obj.BinV Client, Mail

  GetMail = Mail
End Function

```

Data Type BV/V,V

IDL

```

Program 'BinVV' Is
  Define Data Parameter

```

```

1 Client      (BV/V,V)  In
1 Mail        (BV/V,V)  In Out
1 Server      (BV/V,V)  Out
End-Define

```

Visual Basic Code

```

Private Function GetMail() As Variant()
    Dim Client() As Variant
    Dim Mail() As Variant
    Dim Server() As Variant

    Dim ClientWrap() As Variant
    Dim MailWrap() As Variant

    Dim ClientValue() As Byte
    Dim MailValue() As Byte

    ' Define the Bounds of the Arrays
    ReDim Client(8)
    ReDim ClientWrap(12)
    ReDim ClientValue(4)

    ReDim Mail(3)
    ReDim MailWrap(6)
    ReDim MailValue(5)

    ' Setup the Arrays
    For i = 0 To UBound(Client)
        For j = 0 To UBound(ClientWrap)
            For k = 0 To UBound(ClientValue)
                ClientValue(k) = k+j+i+1 ' write any data into the elements
            Next k
            ClientWrap(j) = ClientValue
        Next j
        Client(i) = ClientWrap ' assign the new array to an element of the main array
    Next i

    For i = 0 To UBound(Mail)
        For j = 0 To UBound(MailWrap)
            For k = 0 To UBound(MailValue)
                MailValue(k) = k+j+i+2 ' write any data into the elements
            Next k
            MailWrap(j) = MailValue
        Next j
        Mail(i) = MailWrap ' assign the new array to an element of the main array
    Next i

    obj.BinVVV Client, Mail, Server

    GetMail = Mail
End Function

```

Data Type BV/<nn>,<nn>,<nn>

IDL

```

Program 'BinVnnnnnn' Is
  Define Data Parameter
    1 Client      (BV/3,4,6)    In
    1 Mail        (BV/5,2,5)    In Out
    1 Server      (BV/2,8,2)    Out
  End-Define

```

Visual Basic Code

```

Private Function GetMail() As Variant()
  Dim Client() As Variant
  Dim Mail() As Variant
  Dim Server() As Variant

  Dim ClientValue() As Byte
  Dim MailValue() As Byte

  ' Define the Bounds of the Arrays
  ReDim Client(2,3,5)
  ReDim Mail(4,1,4)

  ' Setup the Arrays
  For i = 0 To UBound(Client, 1)
    For j = 0 To UBound(Client, 2)
      For k = 0 To UBound(Client, 3)
        ReDim ClientValue(9 + i + j + k)
        For m = 0 To UBound(ClientValue)
          ClientValue(m) = i + j + k + m
        Next m
        Client(i, j, k) = ClientValue ' write any data into the elements
      Next k
    Next j
  Next i

  For i = 0 To UBound(Mail, 1)
    For j = 0 To UBound(Mail, 2)
      For k = 0 To UBound(Mail, 3)
        ReDim MailValue(3 + i + j + k)
        For m = 0 To UBound(MailValue)
          MailValue(m) = i + j + k + m
        Next m
        Mail(i, j, k) = MailValue ' write any data into the elements
      Next k
    Next j
  Next i

  obj.BinVnnnnnn Client, Mail, Server

  GetMail = Mail
End Function

```

Data Type BV/V,V,V

IDL

```

Program 'BinVVVV' Is
  Define Data Parameter
    1 Client      (BV/V,V,V)    In
    1 Mail        (BV/V,V,V)    In Out
    1 Server      (BV/V,V,V)    Out
  End-Define

```

Visual Basic Code

```

Private Function GetMail() As Variant()
  Dim Client() As Variant
  Dim Mail() As Variant
  Dim Server() As Variant

  Dim ClientWrap() As Variant
  Dim MailWrap() As Variant

  Dim ClientValue() As Byte
  Dim MailValue() As Byte

  ReDim Client(3)
  For i = 0 To UBound(Client)
    ReDim ClientWrap(i + 1)
    For j = 0 To UBound(ClientWrap)
      ReDim ClientValue(j + 3)
      For k = 0 To UBound(ClientValue)
        ClientValue(k) = i + j + k
      Next k
      ClientWrap(j) = ClientValue
    Next j
    Client(i) = ClientWrap
  Next i

  ReDim Mail(7)
  For i = 0 To UBound(Mail)
    ReDim MailWrap(i + 2)
    For j = 0 To UBound(MailWrap)
      ReDim MailValue(j + 1)
      For k = 0 To UBound(MailValue)
        MailValue(k) = i + j + k
      Next k
      MailWrap(j) = MailValue
    Next j
    Mail(i) = MailWrap
  Next i

  obj.BinVVVV Client, Mail, Server

  GetMail = Server
End Function

```

Data Type Group

IDL

```

Program 'PGroup1' Is
  Define Data Parameter
    1 Data      In Out
    2 Client    (AV)
    2 Mail      (AV)
  End-Define

```

Visual Basic Code

```

Private Sub GetMail()
  obj.PGroup1_Data.Client = "Caroline"
  obj.PGroup1_Data.Mail = "Susan"

  obj.PGroup1 obj.PGroup1_Data

  if obj.PGroup1_Data.Mail <> "Humpty Dumpty" Then
    MsgBox "Humpty Dumpty is not in"
  end if
End Sub

```

Data Type Group/<nn>**IDL**

```

Program 'PGroup2' Is
  Define Data Parameter
    1 Data      (/2) In Out
    2 Client    (AV)
    2 Mail      (AV)
  End-Define

```

Visual Basic Code

```

Private Sub GetMail()
  For i = 0 To UBound(obj.PGroup2_Data)
    obj.PGroup2_Data(i).Client = "Caroline"
    obj.PGroup2_Data(i).Mail = "Susan"
  Next I

  obj.PGroup2 obj.PGroup2_Data

  if obj.PGroup2_Data(0).Mail <> "Humpty Dumpty" Then
    MsgBox "Humpty Dumpty is not in"
  end if
End Sub

```

Data Type Group/<nn> with Nested Group /<nn>**IDL**

```

Program 'PGroup3' Is
  Define Data Parameter
    1 Data      (/2) In Out
    2 Client1   (AV)
    2 Mail1     (AV)
    2 SecondLevel (/1:2) In Out

```

```

    3 Client2      (AV)
    3 Mail2       (AV)
    1 User        (AV)      In Out
End-Define

```

Visual Basic Code

```

Private Sub GetMail()
    Dim User As String

    For i = 0 To UBound(obj.PGroup3_Data)
        obj.PGroup3_Data(i).Client1 = "Caroline"
        obj.PGroup3_Data(i).Mail1 = "Susan"
        For j = LBound(obj.PGroup3_Data(i).SecondLevel) To UBound(obj.PGroup3_Data(i).SecondLevel)
            obj.PGroup3_Data(i).SecondLevel(j).Client2 = "Alice"
            obj.PGroup3_Data(i).SecondLevel(j).Mail2 = "Mary"
        Next j
    Next i

    User = "Minime"

    obj.PGroup3 obj.PGroup3_Data, User

    if obj.PGroup3_Data(0).Mail1 <> "Humpty Dumpty" And _
        obj.PGroup3_Data(0).Mail1.SecondLevel(0).Mail2 = "Humpty Dumpty" Then
        MsgBox "Humpty Dumpty is not in today, but tomorrow"
    end if
End Sub

```

Data Type Group/<nn>,<nn>

IDL

```

Program 'PGroup4' Is
    Define Data Parameter
        1 Data      (/2,3)  In Out
        2 Client   (AV)
        2 Mail     (AV)
    End-Define

```

Visual Basic Code

```

Private Sub GetMail()
    For i = 0 To UBound(obj.PGroup4_Data, 1)
        For j = 0 To UBound(obj.PGroup4_Data, 2)
            obj.PGroup4_Data(i, j).Client = "Caroline"
            obj.PGroup4_Data(i, j).Mail = "Susan"
        Next j
    Next i

    obj.PGroup4 obj.PGroup4_Data

    if obj.PGroup3_Data(0, 0).Mail1 <> "Humpty Dumpty" Then
        MsgBox "Humpty Dumpty is not in today, but tomorrow"
    end if
End Sub

```


Data Type Array Nested in Group

IDL

```
Library 'InpLIB' Is
Program 'PROG' Is
    Define Data Parameter
        1 I          In
        2 inp        (/1:100)
        3 name       (a100)
    End-Define
```

Visual Basic Code

```
Dim Iinpall() As Object
Dim obj As Object
Dim str As String

Set obj = CreateObject("EOL.InpLIB")
obj.ServerAddress = "localhost@RPC/SRV1/CALLNAT"

Iinpall = obj.PROG_i.inp_all

For i = LBound(Iinpall) To UBound(Iinpall)
    Iinpall(i).Name = "test"
Next i

' call the DCOM method
obj.PROG obj.PROG_i

Iinpall = obj.PROG_i.inp_all

For i = LBound(Iinpall) To UBound(Iinpall)
    str = Iinpall(i).Name
Next i
```

Data Type Struct

IDL

```
Struct 'SU' is
    Define Data Parameter
        1 Name (AV)
        1 Age  (I2)
    End-Define

Program 'ConstSU' Is
    Define Data Parameter
        1 Client ('SU') In
        1 Mail   ('SU') In Out
    End-Define
```

Visual Basic Code

```
Private Function GetMail() As Boolean
    Dim ClientSu As Object ' Client structure
    Dim MailSu   As Object ' Mail structure
```

```

Set ClientSu = obj.createStructure_SU
Set MailSu = obj.createStructure_SU

ClientSu.Name = "Caroline"
ClientSu.Age = 21

MailSu.Name = "Susan"
MailSu.Age = 24

obj.ConstSU ClientSu, MailSu

If MailSu.Name <> "Humpty Dumpty" Then
    GetMail = False
Else
    GetMail = True
End If
End Function

```

Data Type Struct/<nn>

IDL

```

Struct 'SU' is
    Define Data Parameter
        1 Name (AV)
        1 Age (I2)
    End-Define

Program 'ConstSUnn' Is
    Define Data Parameter
        1 Client ('SU'/3) In
        1 Mail ('SU'/3) In Out
    End-Define

```

Visual Basic Code

```

Private Function GetMail() As Object()
    Dim ClientSu(2) As Object ' Client structure
    Dim MailSu(2) As Object ' Mail structure

    For i = 0 To UBound(Client)
        Set ClientSu(i) = obj.createStructure_SU
    Next i

    For i = 0 To UBound(MailSu)
        Set MailSu(i) = obj.createStructure_SU
    Next i

    ClientSu(0).Name = "Asterix"
    ClientSu(0).Age = 31
    ClientSu(1).Name = "Automatix"
    ClientSu(1).Age = 34
    ClientSu(2).Name = "Idefix"
    ClientSu(2).Age = 7

    MailSu(0).Name = "Falbala"
    MailSu(0).Age = 19
    MailSu(1).Name = "Gutemine"
    MailSu(1).Age = 50
    MailSu(2).Name = "Cleopatra"

```

```

MailSu(2).Age = 21

obj.ConstSU ClientSu, MailSu
GetMail = MailSu
End Function

```

Data Type Struct/V

IDL

```

Struct 'SU' is
  Define Data Parameter
    1 Name (A40)
    1 Age (I2)
  End-Define

Program 'ConstSUnn' Is
  Define Data Parameter
    1 Client ('SU'/V) In
    1 Mail ('SU'/V) In Out
  End-Define

```

Visual Basic Code

```

Private Function GetMail() As Object()
  Dim ClientSu() As Object ' Client structure
  Dim MailSu() As Object ' Mail structure

  ReDim ClientSu(3) As Object
  ReDim MailSu(2) As Object

  For i = 0 To UBound(Client)
    Set ClientSu(i) = obj.createStructure_SU
  Next i

  For i = 0 To UBound(MailSu)
    Set MailSu(i) = obj.createStructure_SU
  Next i

  ClientSu(0).Name = "Asterix"
  ClientSu(0).Age = 31
  ClientSu(1).Name = "Automatix"
  ClientSu(1).Age = 34
  ClientSu(2).Name = "Idefix"
  ClientSu(2).Age = 7
  ClientSu(3).Name = "Obelix"
  ClientSu(3).Age = 30

  MailSu(0).Name = "Falbala"
  MailSu(0).Age = 19
  MailSu(1).Name = "Gutemine"
  MailSu(1).Age = 50
  MailSu(2).Name = "Cleopatra"
  MailSu(2).Age = 21

  obj.ConstSU ClientSu, MailSu
  GetMail = MailSu
End Function

```

Data Type Struct/V,V

IDL

```

Struct 'SU' is
  Define Data Parameter
    1 Name (A20)
    1 Age (I2)
  End-Define

Program 'ConstSUvv' Is
  Define Data Parameter
    1 Client ('SU'/V,V) In
    1 Mail ('SU'/V,V) In Out
  End-Define

```

Visual Basic Code

```

Private Function GetMail() As Variant()
  Dim ClientSu() As Variant ' Client structure
  Dim MailSu() As Object ' Mail structure
  Dim i as Integer
  Dim j as Integer

  ' helper variable
  Dim ClientSul() As Object
  ReDim ClientSu(3)
  ReDim ClientSul(6)

  For i = 0 To UBound(ClientSu)
    For j = 0 To UBound(ClientSul)
      Set ClientSul(j) = obj.createStructure_SU()
      ClientSul(j).Name = "ClientName_" & Str(j)
      ClientSul(j).Age = j
    Next
    ClientSu(i) = ClientSul
  Next

  ' helper variable
  Dim MailSul() As Object
  ReDim MailSu(5)
  ReDim MailSul(2)
  For i = 0 To UBound(MailSu)
    For j = 0 To UBound(MailSul)
      Set MailSul(j) = obj.createStructure_SU()
      MailSul(j).Name = "MailName_" & Str(j)
      MailSul(j).Age = j+2
    Next
    MailSu(i) = MailSul
  Next
  obj.ConstSUvv ClientSu, MailSu
  GetMail = MailSu
End Function

```

Data Type Struct/V,V,V

IDL

```
Struct 'SU' is
  Define Data Parameter
    1 Name (A20)
    1 Age (I2)
  End-Define

Program 'ConstSUvvv' Is
  Define Data Parameter
    1 Client ('SU2'/V,V,V) In
    1 Mail ('SU2'/V,V,V) In Out
  End-Define
```

Visual Basic Code

```
Private Function GetMail() As Variant()
  Dim ClientSu() As Variant ' Client structure
  Dim MailSu() As Object ' Mail structure

  ' helper variable
  Dim ClientSu1() As Variant
  Dim ClientSu2() As Object
  ReDim ClientSu(3)
  ReDim ClientSu1(6)
  ReDim ClientSu2(2)

  For i = 0 To UBound(ClientSu)
    For j = 0 To UBound(ClientSu1)
      For k = 0 To UBound(ClientSu2)
        Set ClientSu2(k) = obj.createStructure_SU()
        ClientSu2(k).Name = "ClientName_" & Str(k)
        ClientSu2(k).Age = k
      Next
      ClientSu1(j) = ClientSu2
    Next
    ClientSu(i) = ClientSu1
  Next

  ' helper variable
  Dim MailSu1() As Variant
  Dim MailSu2() As Object
  ReDim MailSu(5)
  ReDim MailSu1(2)
  ReDim MailSu2(7)

  For i = 0 To UBound(MailSu)
    For j = 0 To UBound(MailSu1)
      For k = 0 To UBound(MailSu2)
        Set MailSu2(k) = obj.createStructure_SU()
        MailSu2(k).Name = "MailName_" & Str(k)
        MailSu2(k).Age = k+2
      Next
      MailSu1(j) = MailSu2
    Next
    MailSu(i) = MailSu1
  Next
Next
```

```
obj.ConstSU ClientSu, MailSu  
GetMail = MailSu  
End Function
```