Plot tools

Programmable plot

This task can be done with Microsoft Excel. But there are some issues using it: even assuming that all users have access to Excel somewhere, we found that Excel it's expensive about the amount of system resources; many users have Excel with trials addins installed that stops the execution showing messages, between others problems.

Avoiding this kind of problems, this worksheet introduces a Microsoft Chart ActiveX object, which is installed with Windows XP, to drive plots inside Mathcad with a string.

Example using the component

This is a classical Microsoft example:

$T :=$

"Example"	"1st."	"2nd."	"3rd."	"4th."
"East"	15	27.4	22.1	20.4
"West"	30.6	38	28	35
"North"	10	46.9	45	25

The following string drive the MSChart component

$str :=$

$k \leftarrow 0$

$s_{k \leftarrow k+1} \leftarrow$ "ChartArea.ClearFormats"

$s_{k \leftarrow k+1} \leftarrow$ "ChartArea.Font.Bold = False"

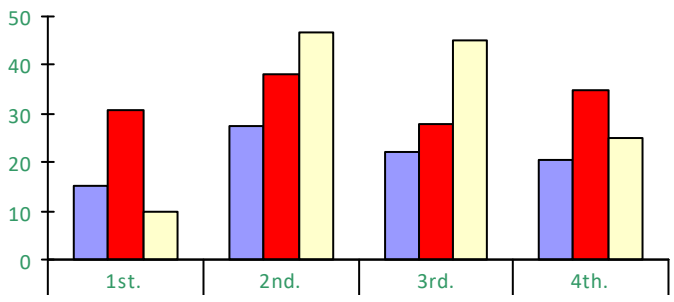
$s_{k \leftarrow k+1} \leftarrow$ "HasDataTable = True"

$s_{k \leftarrow k+1} \leftarrow$ "DataTable.HasBorderOutline = True"

$s_{k \leftarrow k+1} \leftarrow$ "SeriesCollection(2).Interior.Color = vbRed"

s

This is the result



East	15	27.4	22.1	20.4
West	30.6	38	28	35
North	10	46.9	45	25

(str T)

Even this plot can be done in pure Mathcad, probably it's a very hard task. Information about the ActiveX object can be found at

http://msdn.microsoft.com/en-us/library/aa141052(office.10).aspx

http://msdn.microsoft.com/en-us/library/aa171197(office.11).aspx

You can found how to change the plot style and others things. Notice that **str** must to be write without the left dot that you can see in the examples of the above reference.

Further notes

The code inside the component it's the following

```
Set G = MSGraph ' The name of the Component
Set D = G.Application.DataSheet
Set S = G.SeriesCollection

Sub MSGraphEvent_Exec(Inputs,Outputs)
   r = Inputs(1).Rows : c = Inputs(1).Cols
   D.Columns.Delete   ' Take time to do
   For i = 0 To r-1 : For j = 0 To c-1
      D.Range("A1").Item(i,j).Value = Inputs(1).Value(i,j)
   Next : Next
   str = "With G : "
   For i = 0 To Inputs(0).Rows-1
   For j = 0 To Inputs(0).Cols-1
      str = str + "." + Inputs(0).Value(i,j) + " : "
   Next : Next
   ExecuteGlobal str + "End With"
End Sub
```

More sophisticated implementations could be write, but the above code is relevant because shows the essentials.

One very useful instance could be exporting the component as an mcm file (documented under "Saving and Redistributing Components" in the Mathcad help). The code needs to call an ExecuteGlobal visual basic script procedure, which make avaible any kind of vbs code, but in the way that the string is constructed only works inside the MSGraph object, which it's also the Mathcad's object name. More simple: the code seems to be very safe.

Alvaro Díaz, mathcalc@msn.com