

Re: Microsoft Graph to Link to Excel Worksheet

Source:

<http://www.tech-archive.net/Archive/Mac/microsoft.public.mac.office.powerpoint/2004-08/0123.html>

From: Jim Gordon MVP (*goldkey74_at_WarmerThanWarmMail.com*)

Date: 08/15/04

Date: Sun, 15 Aug 2004 13:56:22 -0400

[Cross-posted because Steve & Domenic need the more fame and because this might be of interest to Excel users]

Hi Steve & Domenic,

That's almost perfect!

But after you slap it upside the head there's an error if you're not in Notes or Slide view, so I'm proposing a small modification (see below).

Ok Domenic, I'm going to try to give you a step-by-step description of how to get this to work. Give it a try and let us know how it turns out.

I'm going to assume you've never worked with visual basic. Although there are a lot of steps to take, none of them are hard or complicated. You may discover some new and interesting things about Office along the way. These directions are for Office 2004.

You should have two existing files to work with: The Excel workbook you want to use as the source of the data for the graph, and a presentation file (in this example it's cleverly called "Presentation 1.")

Begin by making the visual basic editor easy to get at in both Excel and PowerPoint. Open Excel and PowerPoint. In each application turn on the visual basic toolbar by using View > Toolbars > Visual Basic. The toolbar looks a little different in Excel and PowerPoint, but don't worry about the different look. Don't do anything with the toolbar for the moment. I had you turn it on as a convenience for later on.

The next thing to do is to open the workbook that has the graph in it that you want to use in PowerPoint. I presume you've done some customization to the chart's formatting and have it "just right." Let's now preserve that formatting so we can use it later on.

Click once on the chart to select it. You can tell the chart is selected when all the outermost corners have little black "handles" available.

When the chart is selected use the Chart menu (it's only available when the chart is selected) and choose "Chart Type" (the first option). Click the "Custom Types" tab. In the lower left section where it says "Select From" choose "User-Defined." Click the "Add" button. Give your formatting a name and description then click "OK." Now you've saved your customizations and can use them on any future charts. Click "OK" to close the Chart Type dialog box. Yes, this method can be used to build a library of your favorite chart formats.

We're done with Excel for the moment, so Apple+Q to quit Excel.

Open PowerPoint to blank slide presentation.

From PowerPoint's Insert menu choose Object > Microsoft Excel Chart

Excel will open and display a default workbook called "Chart in Presentation 1" showing a graph worksheet called "graph1" and the source data for the graph on "sheet1" (look at the sheet tabs at the bottom of the workbook). Click the "sheet1" tab. Drag your mouse over all of the data so that everything on the worksheet is highlighted, then from the Edit menu choose Clear > All. This will result in an empty worksheet. Click into Cell A1 so that all you have is an empty worksheet with cell A1 selected.

Now go to the File menu and open the workbook that has the graph you want to use. I will refer to this workbook as "the Source." It should be the first item in the recently used files if you have been following along with this.

In the Source workbook, find the cells that have the data that the chart is using and select (highlight) all of them, then use Edit > Copy (or Apple+C). From Excel's "Window" menu switch to the "Chart In Presentation 1" workbook. Then use Edit > Paste (or Apple+v). Click the widget when it appears. In the widget click "Link Cells." Now the workbook in PowerPoint is linked to the data range in the Source workbook.

Next, you will use the graph format you saved earlier. Click on the "graph1" tab of the "Chart in Presentation 1" workbook. From the Chart menu choose "Chart Type." On the "Custom Types" tab choose "User Defined" in the "Select From" section. Choose the chart type that you saved earlier then click "OK." The graph should look identical to the one in the Source workbook.

From the "Window" menu switch to the Source workbook. Close the source workbook (Apple+W, File>Close, or click the red Close button). That should bring the "Chart in Presentation 1" workbook to the front.

On the visual basic toolbar click the "Record New Macro" button. Click OK (for now just take the defaults, but you can give your Macro a name or description if you want to). The macro recorder will now record the steps you take.

From the Edit menu choose Links>Update Now then click the "Close" button. Click the "Record New Macro" button again to stop the recording (this button is a toggle switch).

Now click the "Visual Basic Editor" button on the visual basic toolbar. You will see the editor has 3 windows: "Project", "Properties", and "Chart in Presentation 1 – Module 1 (Code)." If you only see 2 windows, in the Project window click the disclosure triangles for "VBAProject (Chart in Presentation 1)" to reveal the "Modules" folder and then in that folder double-click "Module1" to display the module window.

This is what I got when I followed these steps although the newsgroup reader will probably alter the format a little. I removed the underscores (they're not needed but you don't have to remove them):

```
Sub Macro1()  
'  
' Macro1 Macro  
' Macro recorded 8/15/2004 by James Gordon  
'  
'  
    ActiveWorkbook.UpdateLink Name:=  
"PB:Users:jamesgordon:Desktop:Graph example:SourceWorkbook.xls", Type:=  
xlExcelLinks  
  
End Sub
```

Think of the Module window as a word processor, except that it uses visual basic as its dictionary instead of English or some other spoken language.

If you read this module from the top down you'll see it starts with the word "Sub" followed by a space and then the name of the subroutine, in this case it's "Macro1". You also need the parenthesis (don't worry about what they are for – we're not going to use them).

A few lines start with apostrophes. Any line that starts with an apostrophe is just a comment and is not processed.

The line of code that does all the work starts with Activeworkbook.UpdateLink... followed by the path name to the Source workbook.

"End Sub" is the end of the subroutine.

When this subroutine is run it causes Excel to update the data range in the Presentation's workbook so that it exactly matches the data from the Source workbook.

There's one additional thing that needs to be added to make it more automatic. Just before "End Sub" add this line of text on its own line:
workbooks.close

We're done with the Visual Basic Editor. From the "Excel" menu choose Close and Return to Microsoft Excel (or use Apple+Q).

It is necessary to Save the Excel workbook at this point. Use Apple+S, File > Save, or click the "Save" button on Excel's standard toolbar. Do NOT use Save-As.

Make sure that "chart1" worksheet is selected so that you see the graph. Now we're done with the Excel workbook, so click the red close button (or Apple+W or File>Close and Return to Presentation 1).

You should now be in the presentation and see the graph.

PowerPoint does not have the ability to record macros, so we will have to do a little bit of manual labor now.

Click the Visual Basic Editor button from the Visual Basic toolbar in PowerPoint. When the editor opens, from the menu choose Insert > Module.

This is where Steve's Macro comes into play. Copy the following subroutine beginning with "Sub" all the way through and including "End Sub" and paste it into the editor:

```
Sub StevesOpener()  
Dim oSl As Slide  
Dim oSh As Shape  
On Error GoTo errorhandler  
For Each oSl In ActivePresentation.Slides  
    For Each oSh In oSl.Shapes  
  
        ' is it an embedded OLE object?  
        If oSh.Type = msoEmbeddedOLEObject Then  
            ' crude check: is it a chart?  
            If InStr(UCase(oSh.OLEFormat.ProgID), "CHART") > 0 Then  
                ' slap it upside the head  
                oSh.OLEFormat.Activate  
                ActiveWindow.Selection.Unselect  
            End If  
        End If  
    Next oSh  
Next oSl  
Exit Sub  
errorhandler:  
    'Get around the problem of not being in slide or notes view  
    Resume Next  
End Sub
```

That's all we need the visual basic editor for, so close the editor from the "PowerPoint" menu by choosing "Close and return to Microsoft PowerPoint" (or use Apple+Q).

At this point it is important to save the presentation using Apple+S, File>Save, or click the Save button on PowerPoint's standard toolbar.

We're Done!

Just to try things out, close Excel and PowerPoint completely.

Now open the Source workbook and make some changes to the data range. Save the workbook and close Excel completely.

Next, open the Presentation. You'll probably be prompted about Macros to which you should click "Enable Macros." On the visual basic toolbar click the "Run Macro" button. "StevesOpener" should be selected so all you have to do is click the "Run" button. You may be again prompted about Macros to which you should click "Enable Macros." Excel will then open. Click Excel's "Run Macro" button and run Macro1. The workbook will update and return to PowerPoint.

There are two more automations that could be possible. PowerPoint could call the Excel Macro so you don't have to manually run Macro1 in Excel. The other is that the PowerPoint Macro could be made to run automatically when the presentation is opened (you would need to turn it into an Auto_Open PowerPoint add-in).

I'll let others who want to jump in offer their suggestions on how to do those things.

-Jim

```
--
Jim Gordon
Mac MVP
MVP FAQ
<http://mvp.support.microsoft.com/default.aspx?scid=fh;EN-US;mvpfaqs>
Steve Rindsberg wrote:
> Try:
>
> Sub ActivateEmAllDanO()
>
> Dim oSl as Slide
> Dim oSh as Shape
>
> for each oSl in ActivePresentation.Slides
>   for each oSh in oSl.Shapes
>
>     ' is it an embedded OLE object?
>     if osh.type = msoembeddedOLEobject then
>       ' crude check: is it a chart?
>       if InStr(Ucase(osh.oleformat.progid),"CHART") > 0 then
>         ' slap it upside the head
>         osh.oleformat.activate
```

microsoft.public.mac.office.powerpoint: Re: Microsoft Graph to Link to Excel Worksheet

```
>         activewindow.selection.unselect
>     end if
> end if
> next oSh
> Next oSl
> End Sub
>
> In article <eJZ3ddhgEHA.644@tk2msftngp13.phx.gbl>, Jim Gordon MVP wrote:
>
>>I'm not quite ready to throw in the towel on this one.
>>
>>It seems to me there should be a macro that would handle the activation
>>of the chart object for you.
>>
>>I scarfed around PPT's help and found this code sample but it does not
>>work. But it seems to me with the proper syntax this could be made to
>>work with an auto_open macro that would cause the embedded object to be
>>activated so that it would update itself.
>>
>>Perhaps someone wiser about this will offer a code snippet.
>>
>>Sub Updater()
>>With ActivePresentation.Slides(1).Shapes(1)
>>    For Each sVerb In OLEFormat.ObjectVerbs
>>        If sVerb = "Open" Then
>>            With Application.ActionSettings(ppMouseClick)
>>                .Action = ppActionOLEVerb
>>                .ActionVerb = sVerb
>>            End With
>>        Exit For
>>    End If
>>    Next
>>End With
>>End Sub
>>
>
> --
> Steve Rindsberg, PPT MVP
> PPT FAQ:  www.pptfaq.com
> PPTools:  www.pptools.com
> =====
> Featured Presenter, PowerPoint Live 2004
> October 10-13, San Diego, CA  www.PowerPointLive.com
> =====
>
```