Using the ODS Graphics Designer to Create Your Own Templates in SAS® 9.2

Philip R Holland, Holland Numerics Ltd

Agenda

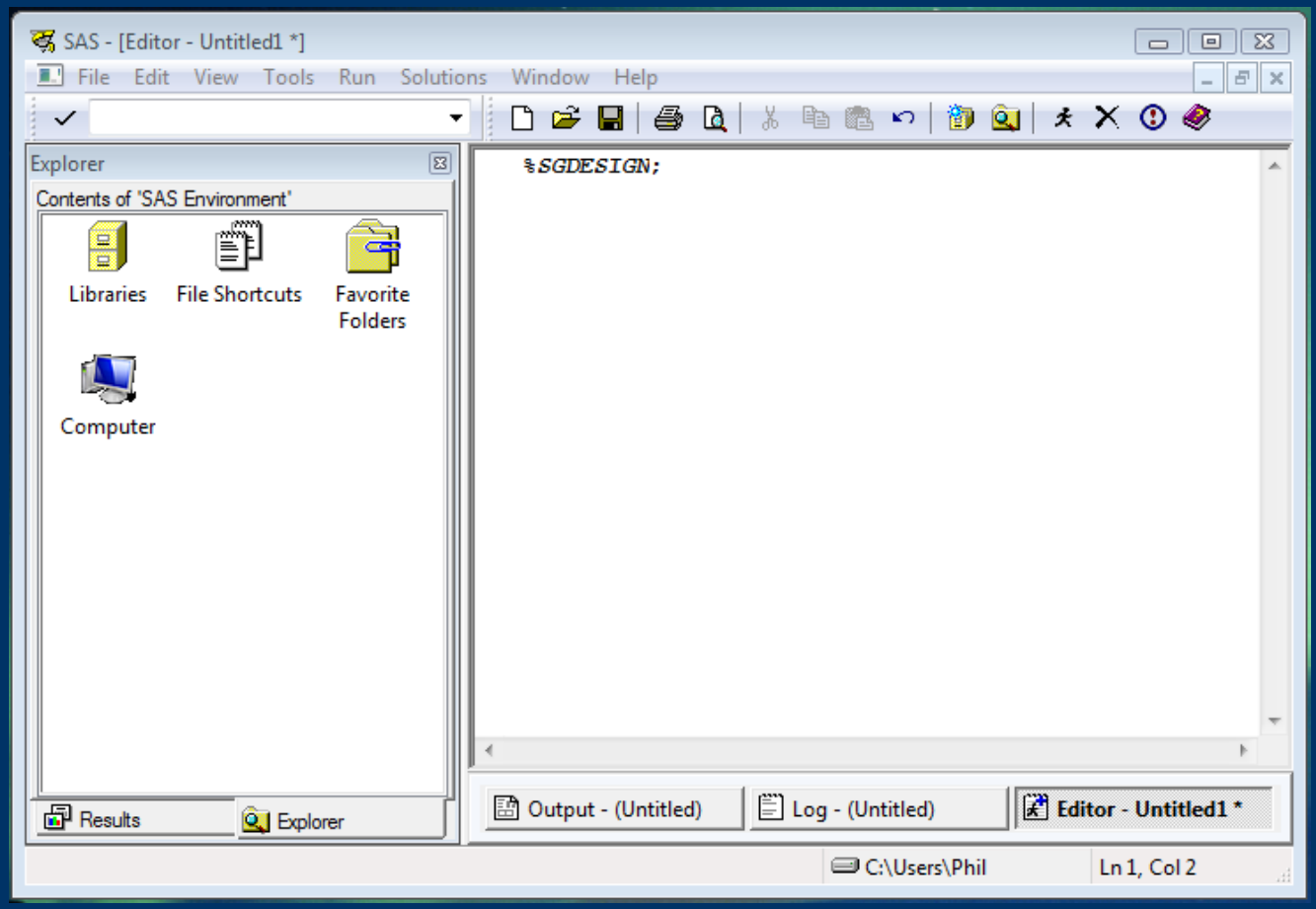
- Introduction to the ODS Graphics Designer
- Using the Gallery to create simple templates
- Building a template from a blank graph
- Saving the template as a Designer file (*.sgd)
- Saving the template as a SAS program (*.sas)
- Summary and conclusions

Introduction to the ODS Graphics Designer - 1

- Pre-production release of the ODS Graphics Designer in SAS 9.2 Phase 2.
- GUI interface for building ODS templates for graphs.
- Java 1.5 application, so compatible with Windows, UNIX and Linux.
- Will build a basic ODS Template for almost all types of graphs.

Introduction to the ODS Graphics Designer - 2

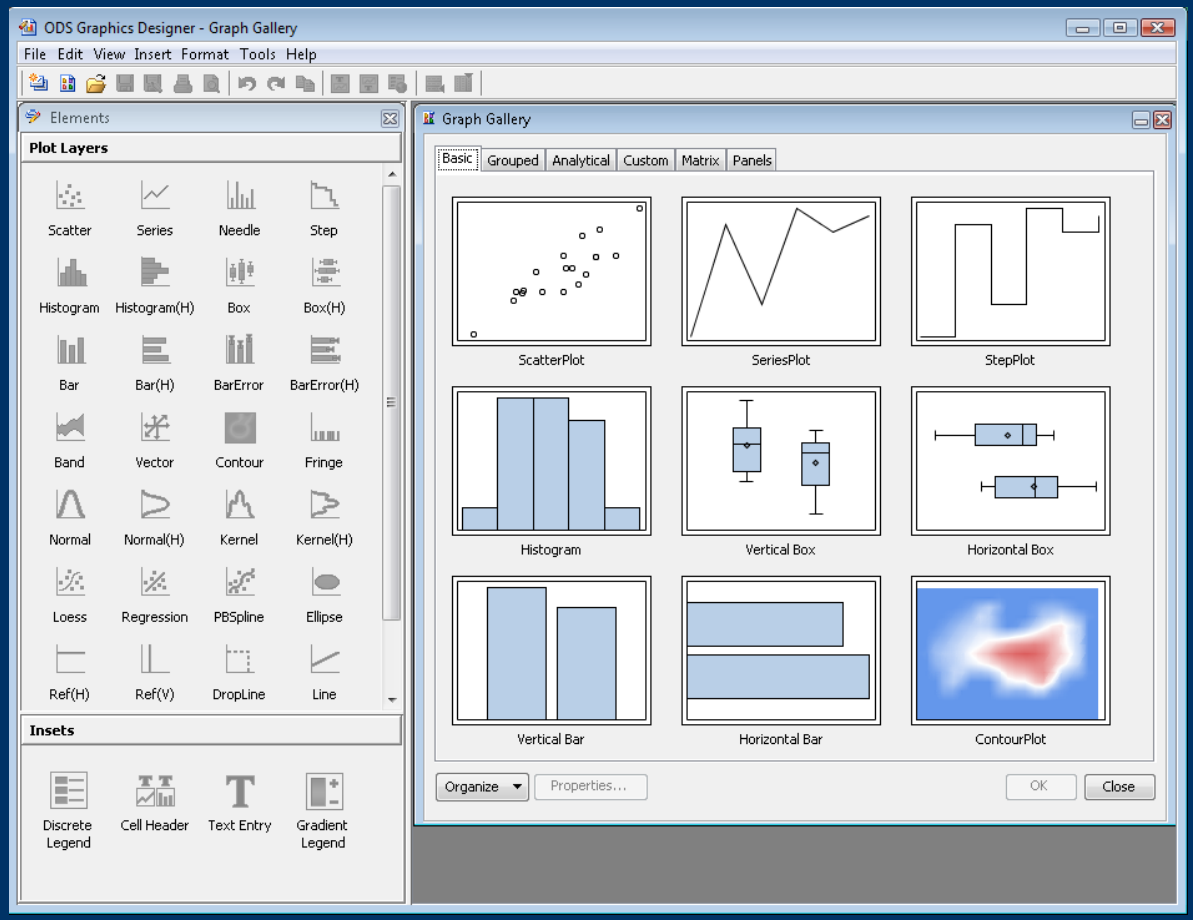
The ODS Graphics Designer can be started from within SAS 9.2 with a macro call.



Introduction to the ODS Graphics Designer - 3

This starts a Java application which can be used to build ODS Graphics templates, and then send them back to SAS for execution.

The initial screen layout includes Elements and Graph Gallery.

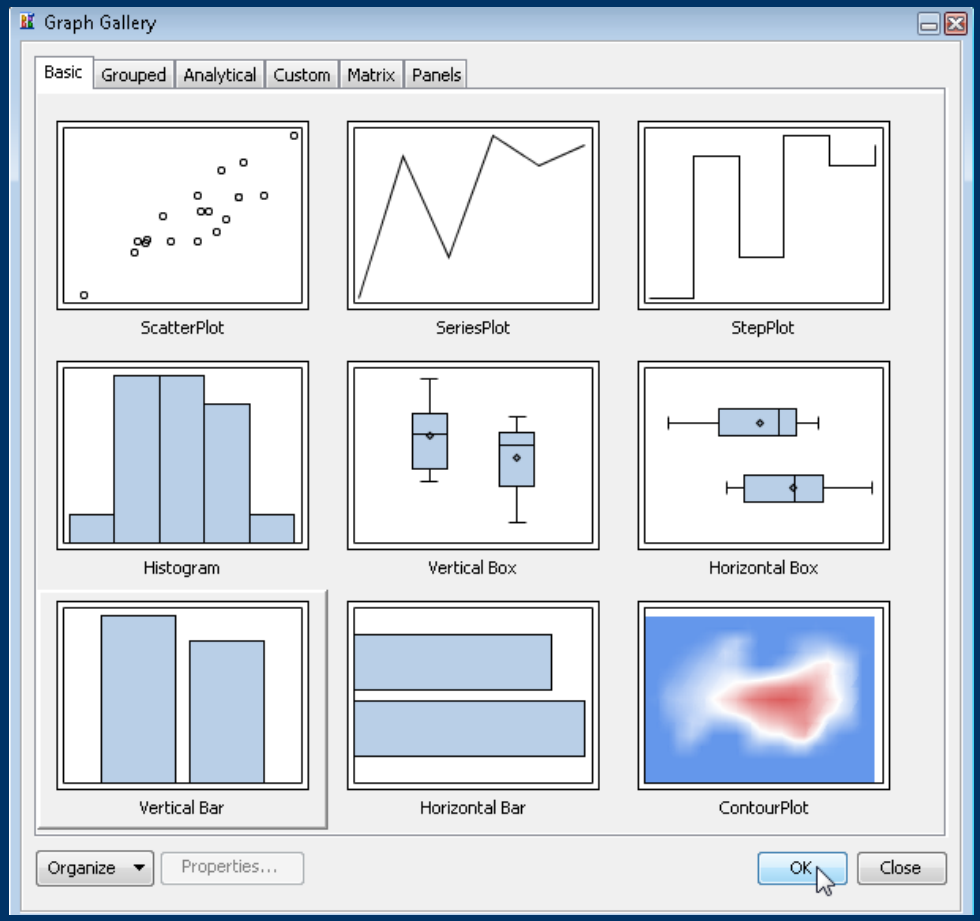


Introduction to the ODS Graphics Designer - 4

As the ODS Graphics Designer is not part of the SAS System, but an external program using the SAS software environment, the SAS data sets used to create templates must be stored as permanent SAS data sets in accessible libraries.

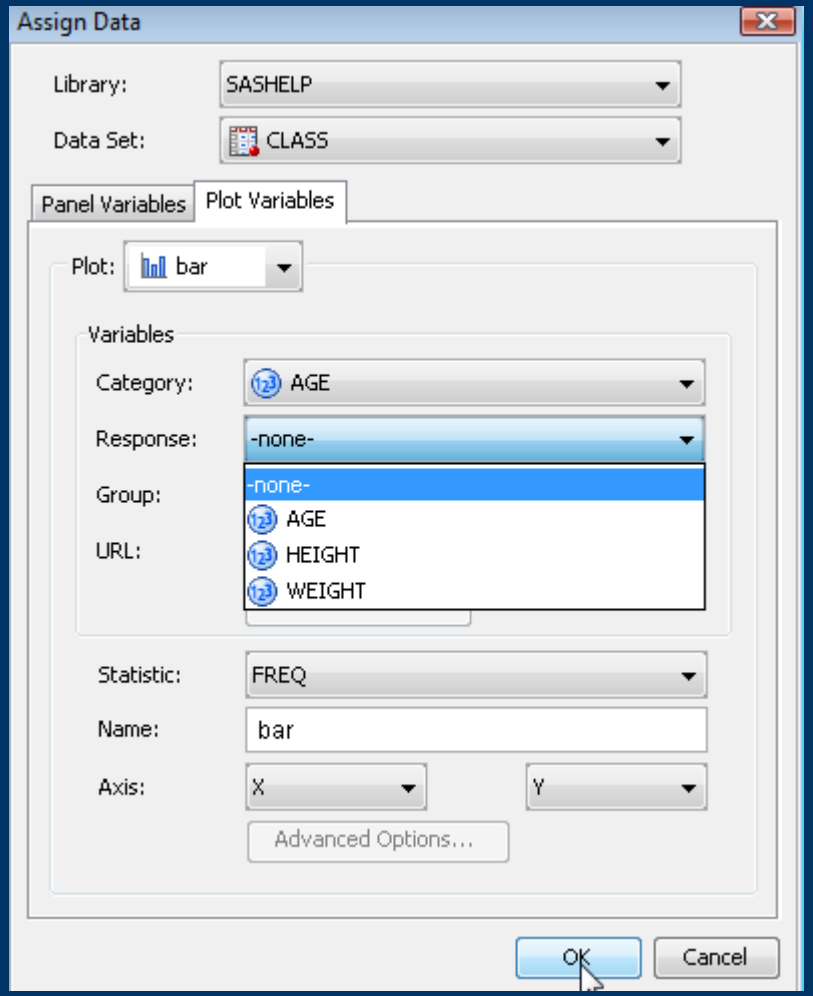
Using the Gallery to create simple templates - 1

The Gallery provides a collection of typical graphical reports that can be used as starting points for more complex reports, but can be used with minimal customisation to generate simple templates.



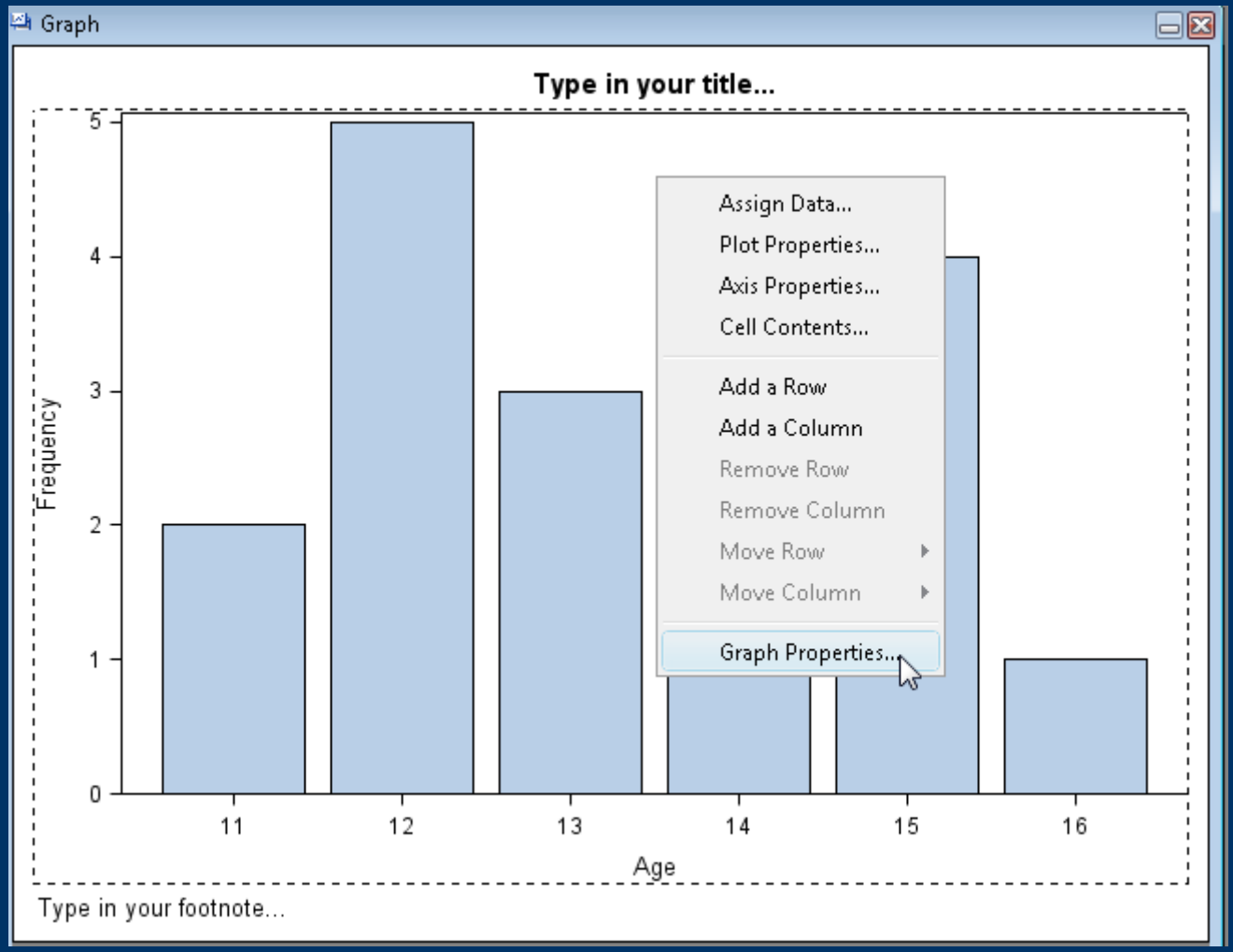
Using the Gallery to create simple templates - 2

Once a Gallery entry has been selected you will be asked for details of the data to be plotted.



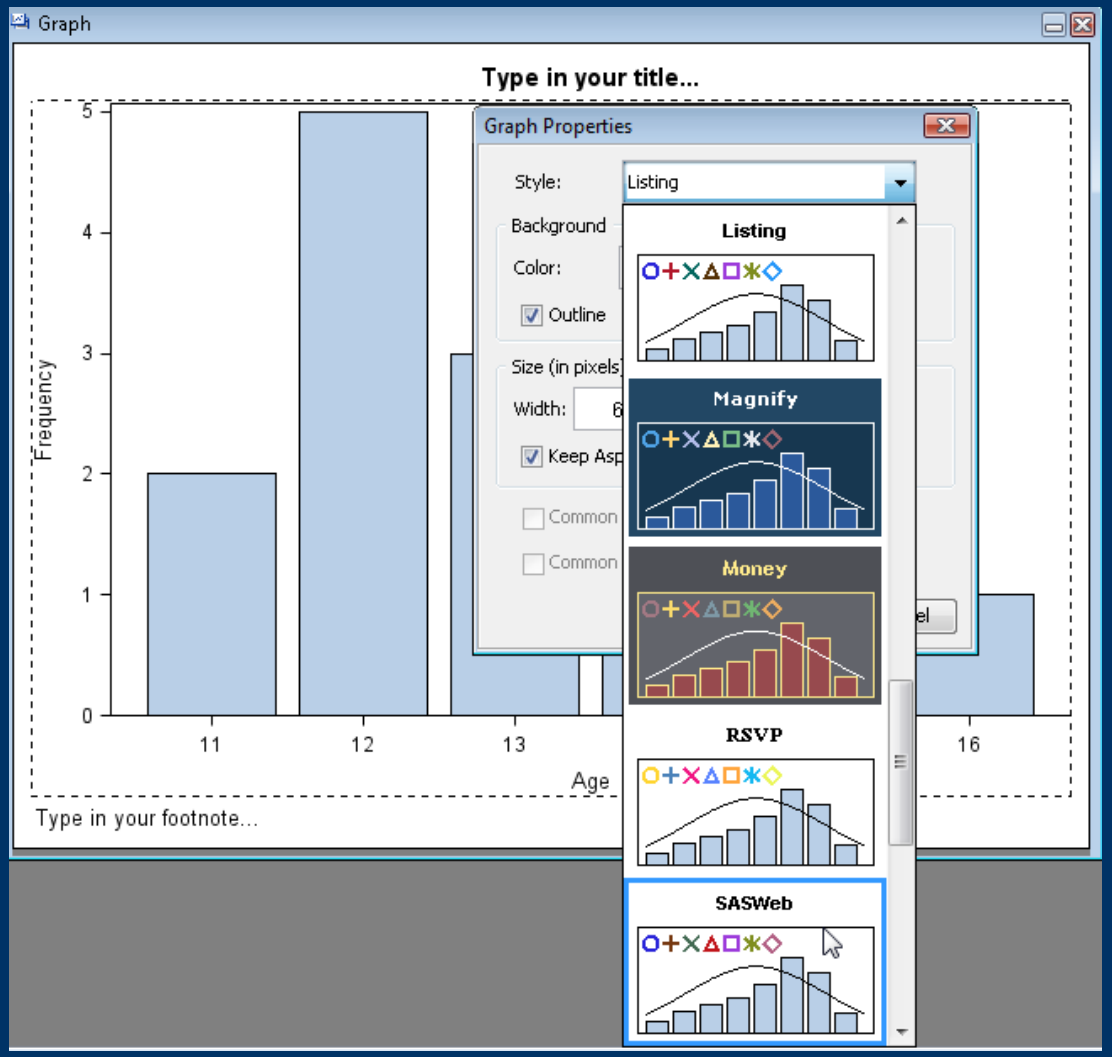
Using the Gallery to create simple templates - 3

The style and graph layout can also be selected.



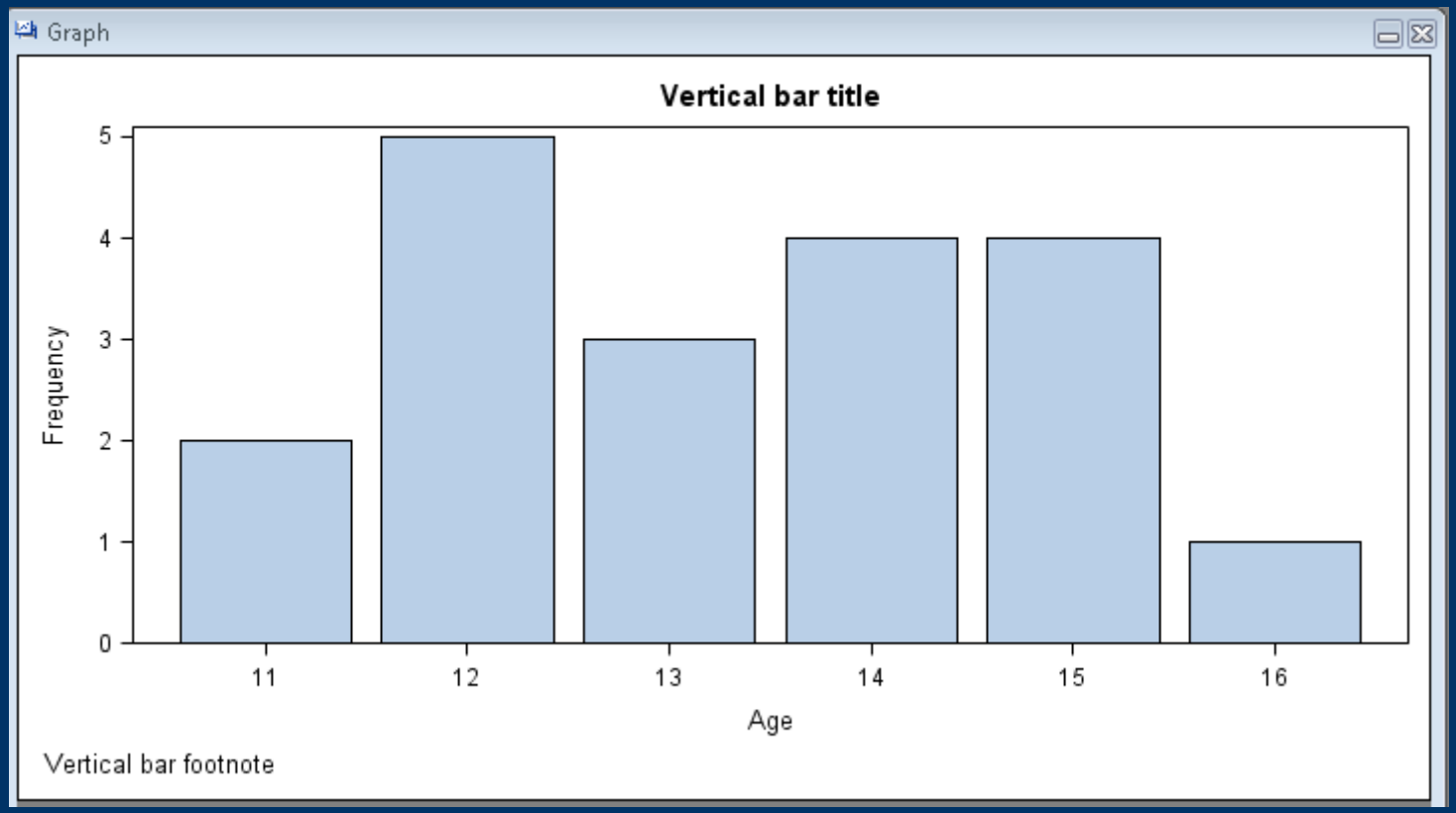
Using the Gallery to create simple templates - 4

The Graph Properties option is used to change the style, background and image size.



Using the Gallery to create simple templates - 5

The graph and the code can be viewed together, if the Code window is not already visible, by clicking on the View menu option and selecting Code.



Using the Gallery to create simple templates - 6

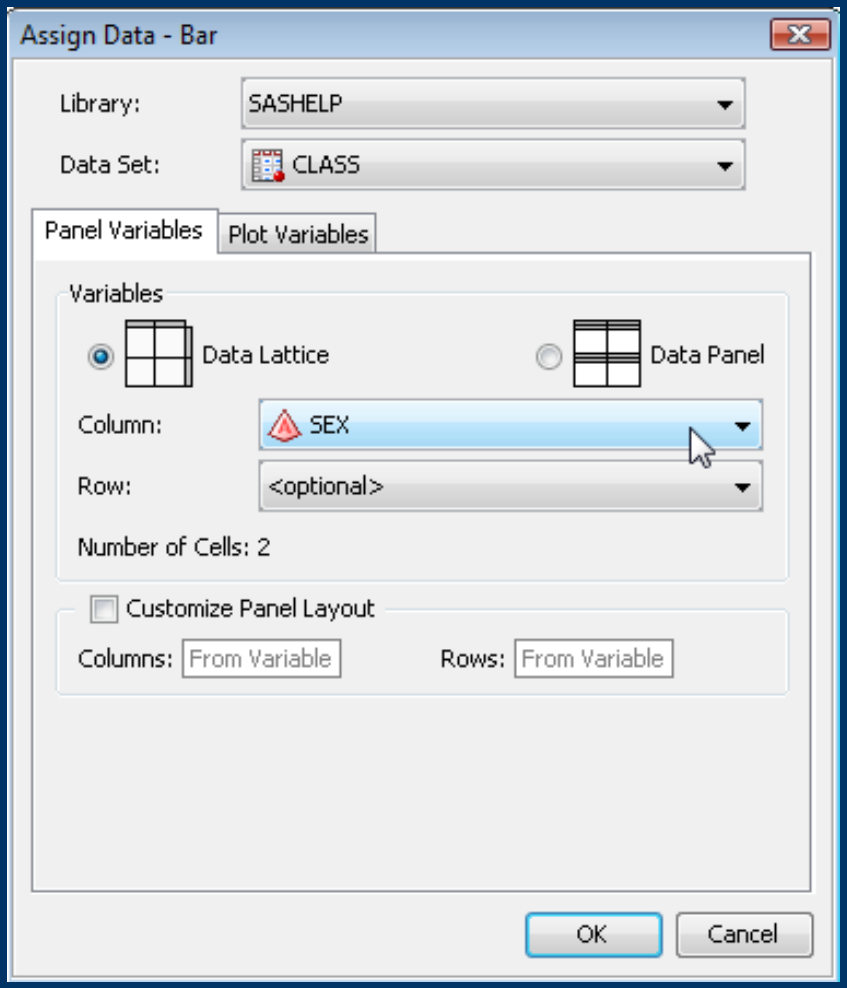
Note that, because I had not explicitly specified the size of the image, changing the size of the Graph window will automatically change the values of DESIGNWIDTH= and DESIGNHEIGHT= on the BEGINGRAPH statement.

Code - Graph

```
proc template;
define statgraph sgdesign;
dynamic _AGE;
begingraph / designheight=376 designwidth=712;
entrytitle _id='title' halign=center 'Vertical bar title';
entryfootnote _id='footnote' halign=left 'Vertical bar footnote';
layout lattice _id='lattice' / columndatarange=data columngutter=10 rowdatarange=data rowgutter=10;
      layout overlay _id='overlay';
        barchart _id='bar' x=_AGE y=_HEIGHT / name='bar' stat=freq;
      endlayout;
endlayout;
endgraph;
end;
run;
```

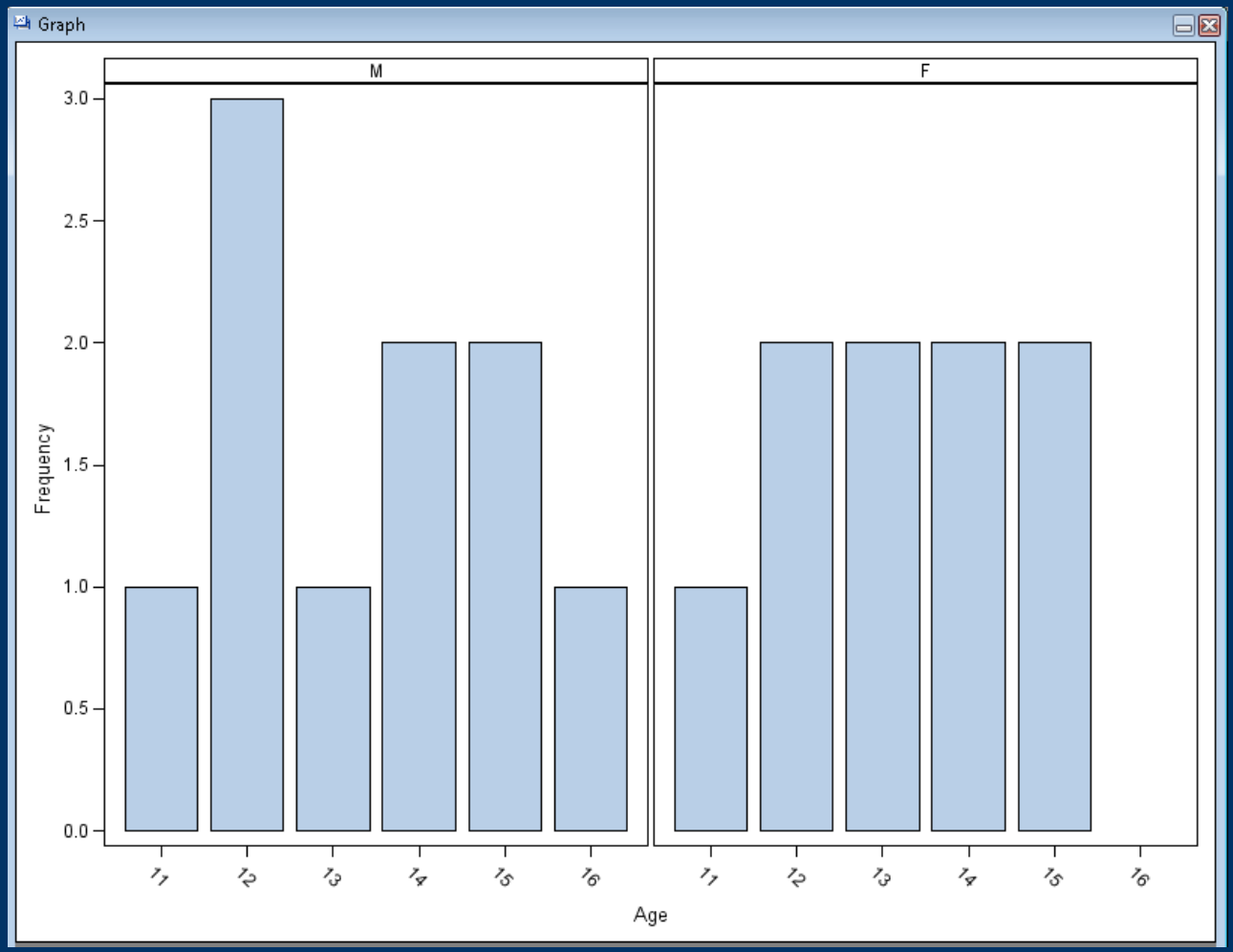
Using the Gallery to create simple templates - 7

There is also an opportunity to split the graph into a panelled layout using the Assign Data option and specifying the Panel Variables.



Using the Gallery to create simple templates - 8

This will create the following graph, ...



Using the Gallery to create simple templates - 9

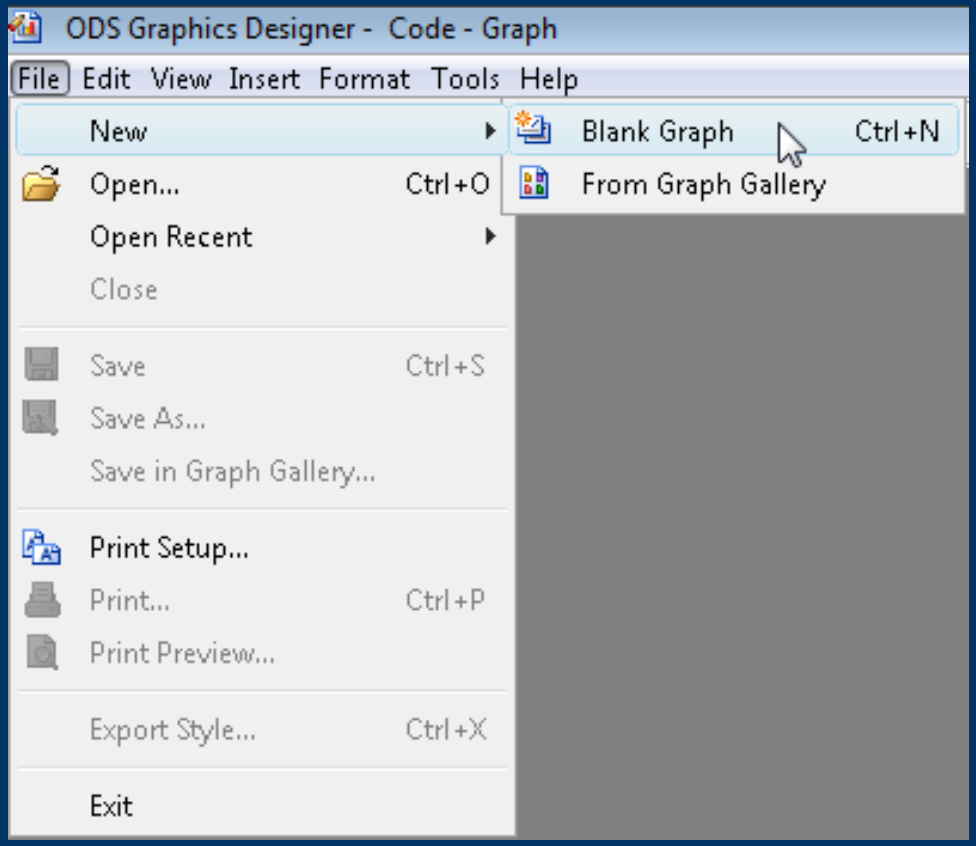
… with the corresponding code for the template.

Code - Graph

```
proc template;
define statgraph sgdesign;
dynamic _panelnumber_;
begingraph / designheight=600 designwidth=800;
layout datalattice _id='datalattice' columnvar=SEX / panelnumber=_panelnumber_ cellheightmin=1
cellwidthmin=1 columnaxisopts=(type=discrete) columndatarange=unionall columngutter=3 columns=2
headerlabeldisplay=value rowdatarange=unionall rowgutter=3 rows=1;
        layout prototype _id='prototype' / ;
          barchart _id='bar' x=AGE / barwidth=0.85 name='bar' stat=freq;
        endlayout;
endlayout;
endgraph;
end;
run;
```

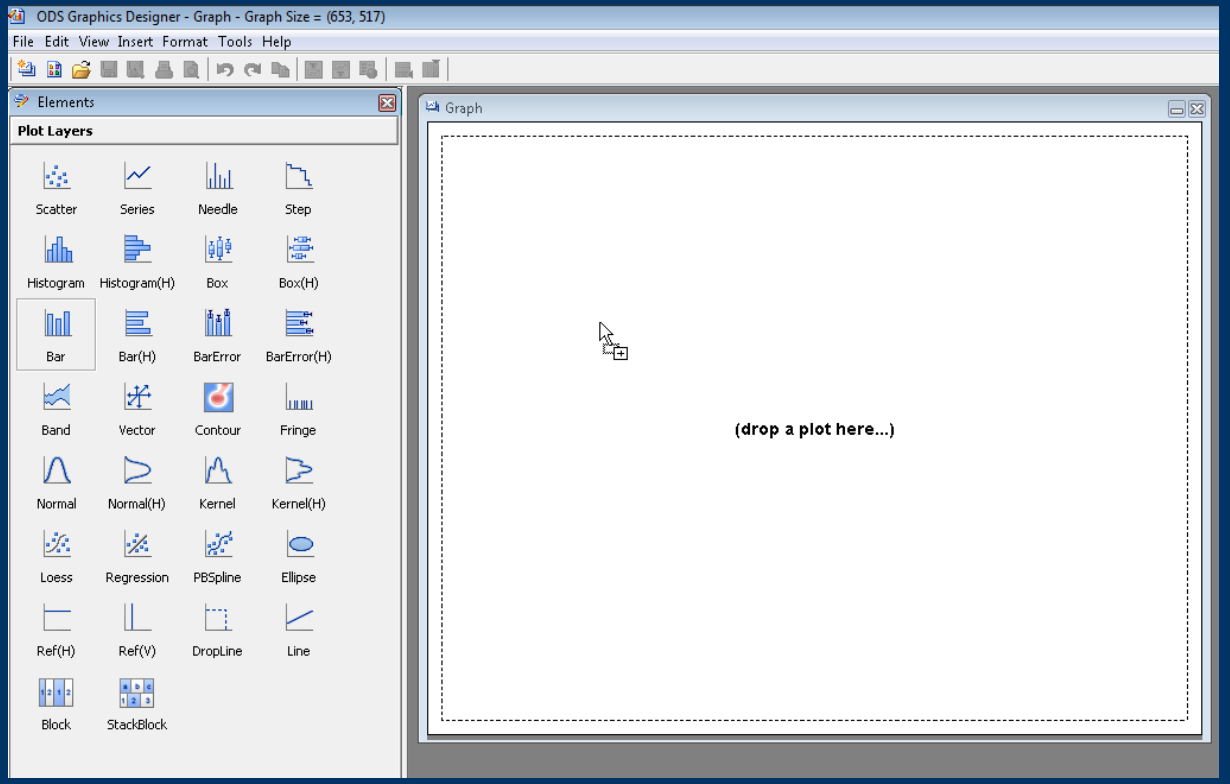
Building a template from a blank graph - 1

The difference between a Gallery entry and a blank graph is that no defaults have been set, so you have much more control over the content and layout of the final template.



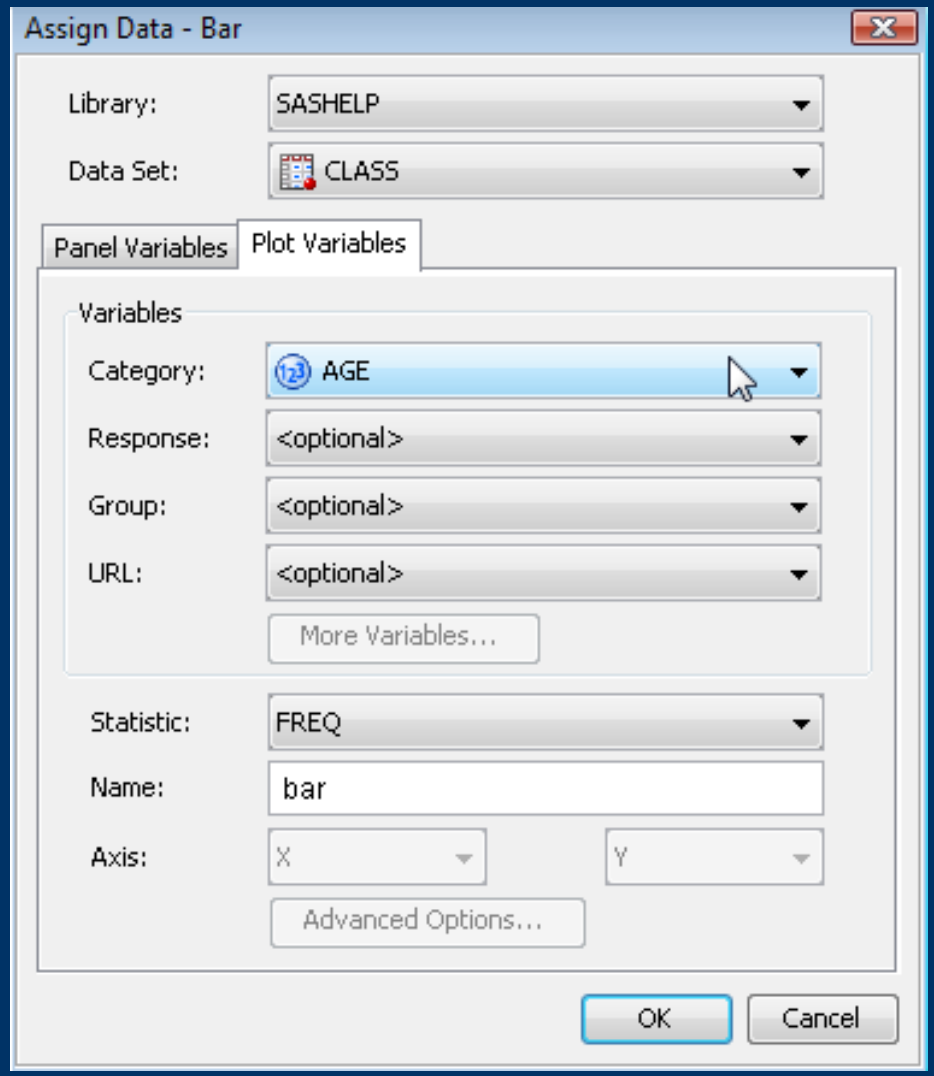
Building a template from a blank graph - 2

A blank graph is created ready for you to add what you need. It is now straightforward to add multiple graphical objects that use the same data to a single graphical report. You just drag the Elements from the appropriate group onto the graph area.



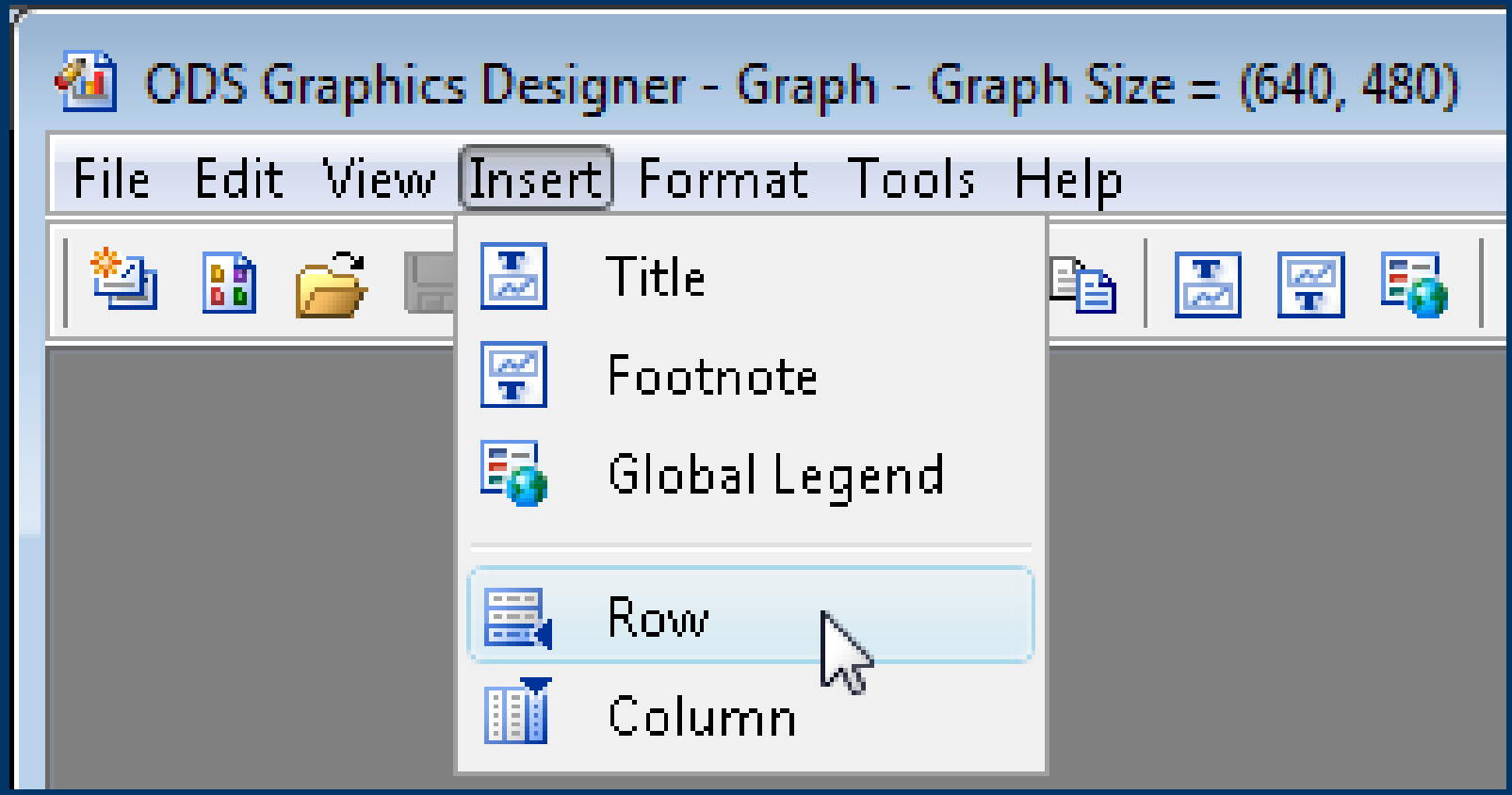
Building a template from a blank graph - 3

Once a plotting Element has been dragged the SAS data is requested.



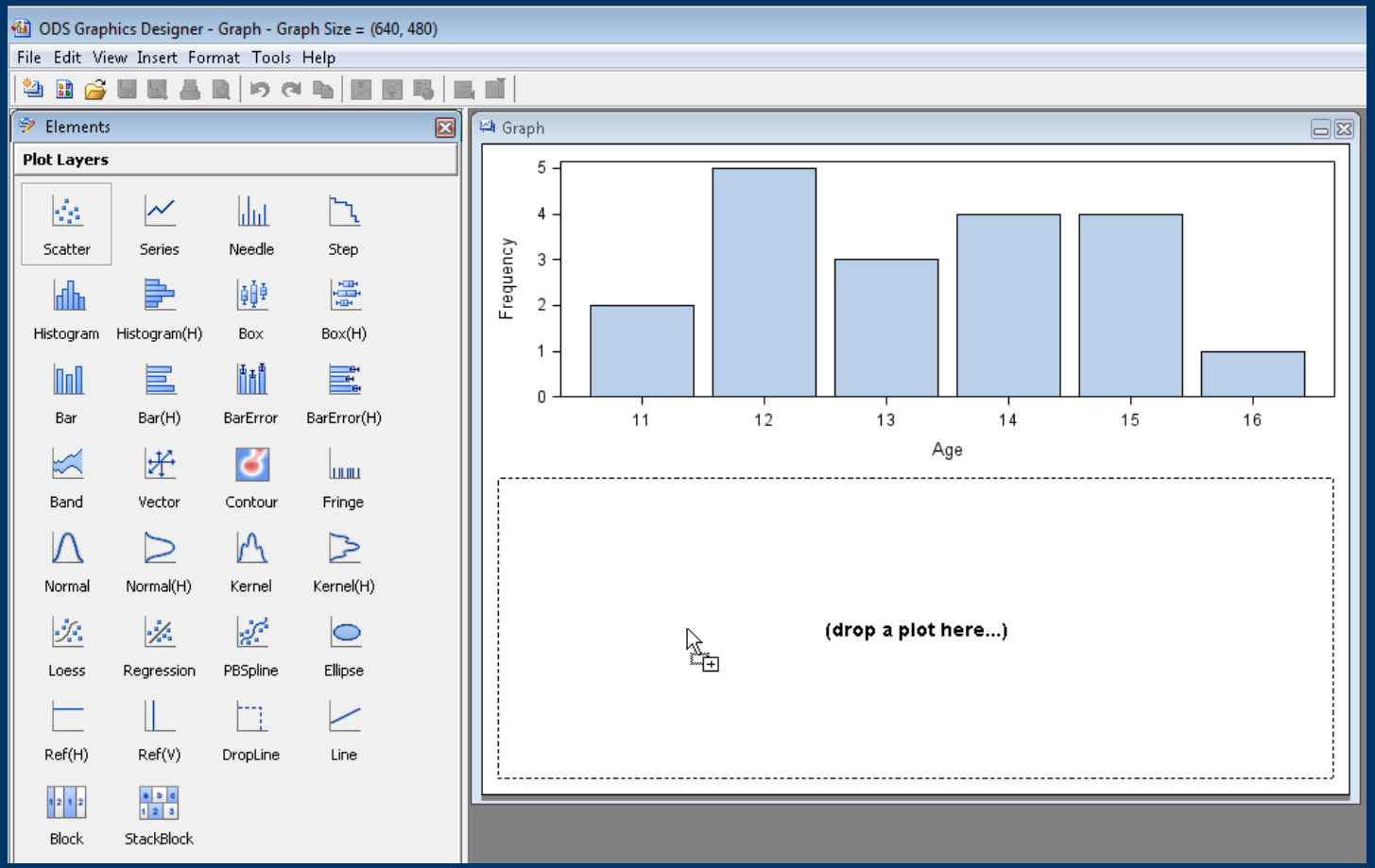
Building a template from a blank graph - 4

An additional row for another graph on the same page can be created.



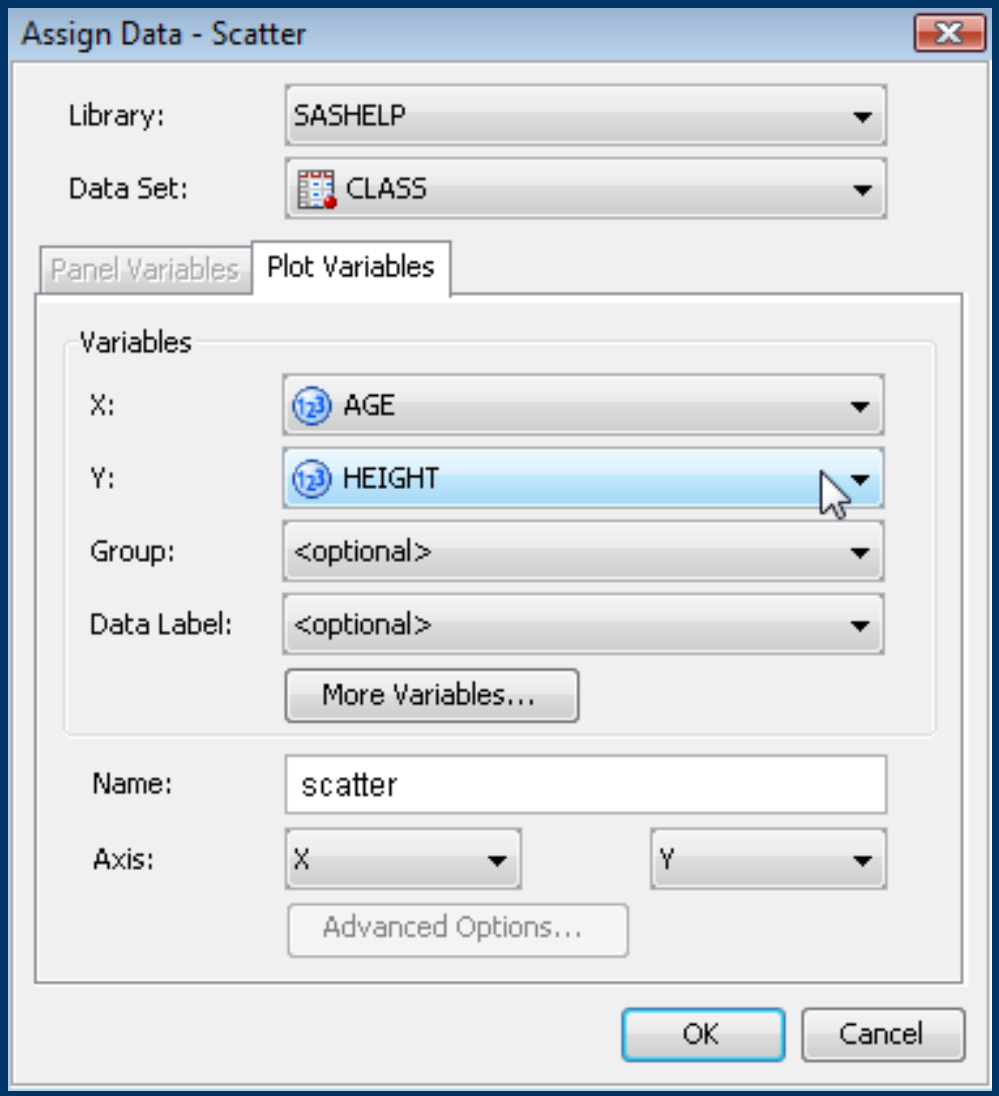
Building a template from a blank graph - 5

This generates a blank row in the image ready for another graph to be dragged into place.



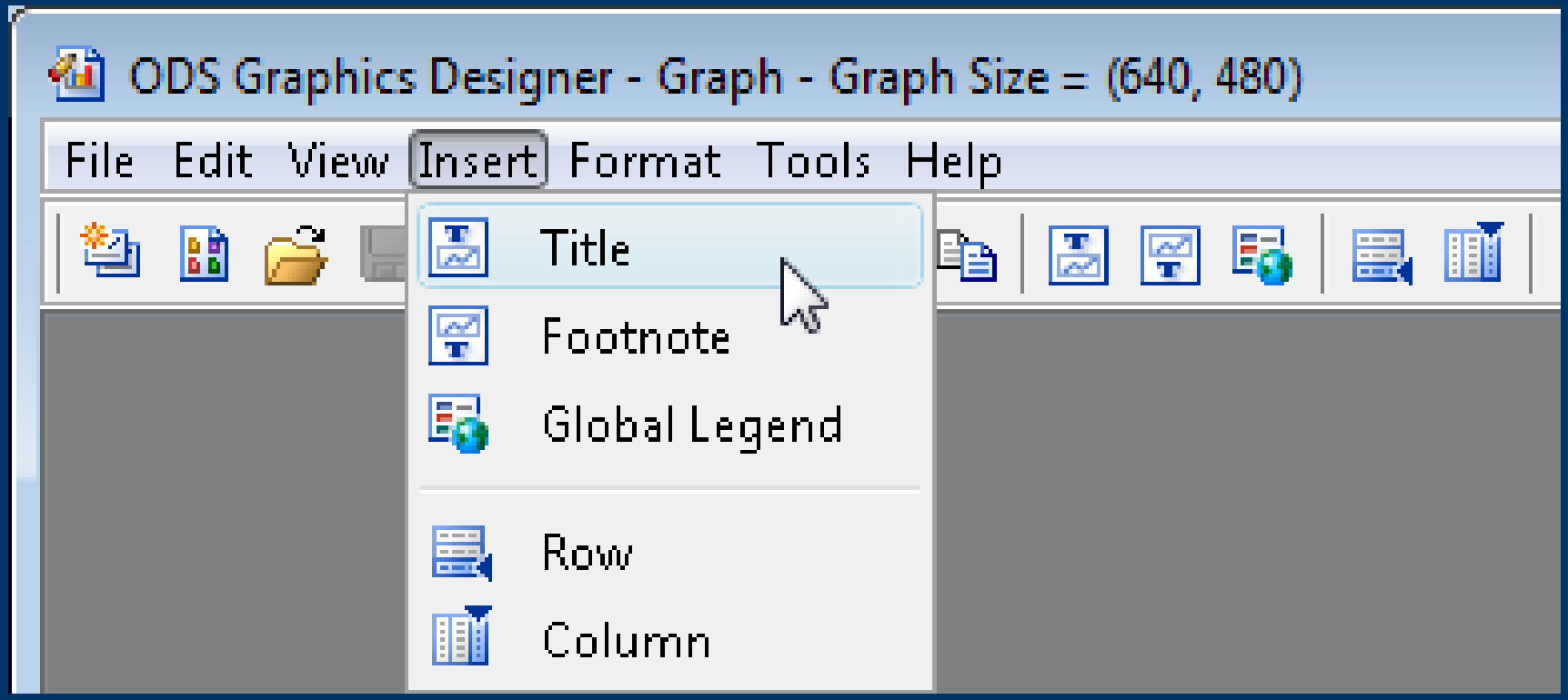
Building a template from a blank graph - 6

The data for the new graph is automatically requested.



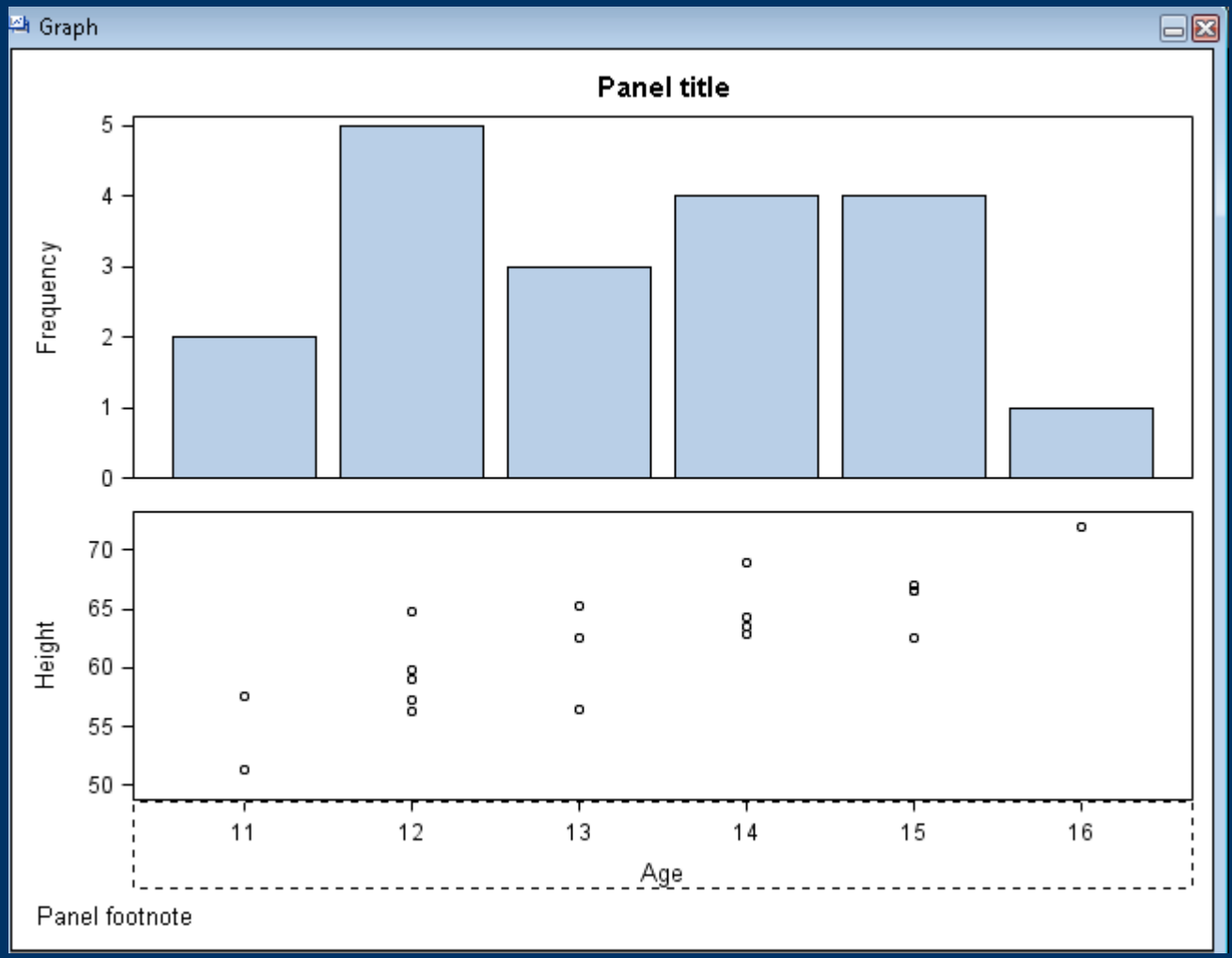
Building a template from a blank graph - 7

Titles and footnotes can now be added.



Building a template from a blank graph - 8

The graph is now ready for use, ...



Building a template from a blank graph - 9

… along with the SAS Program for the template.

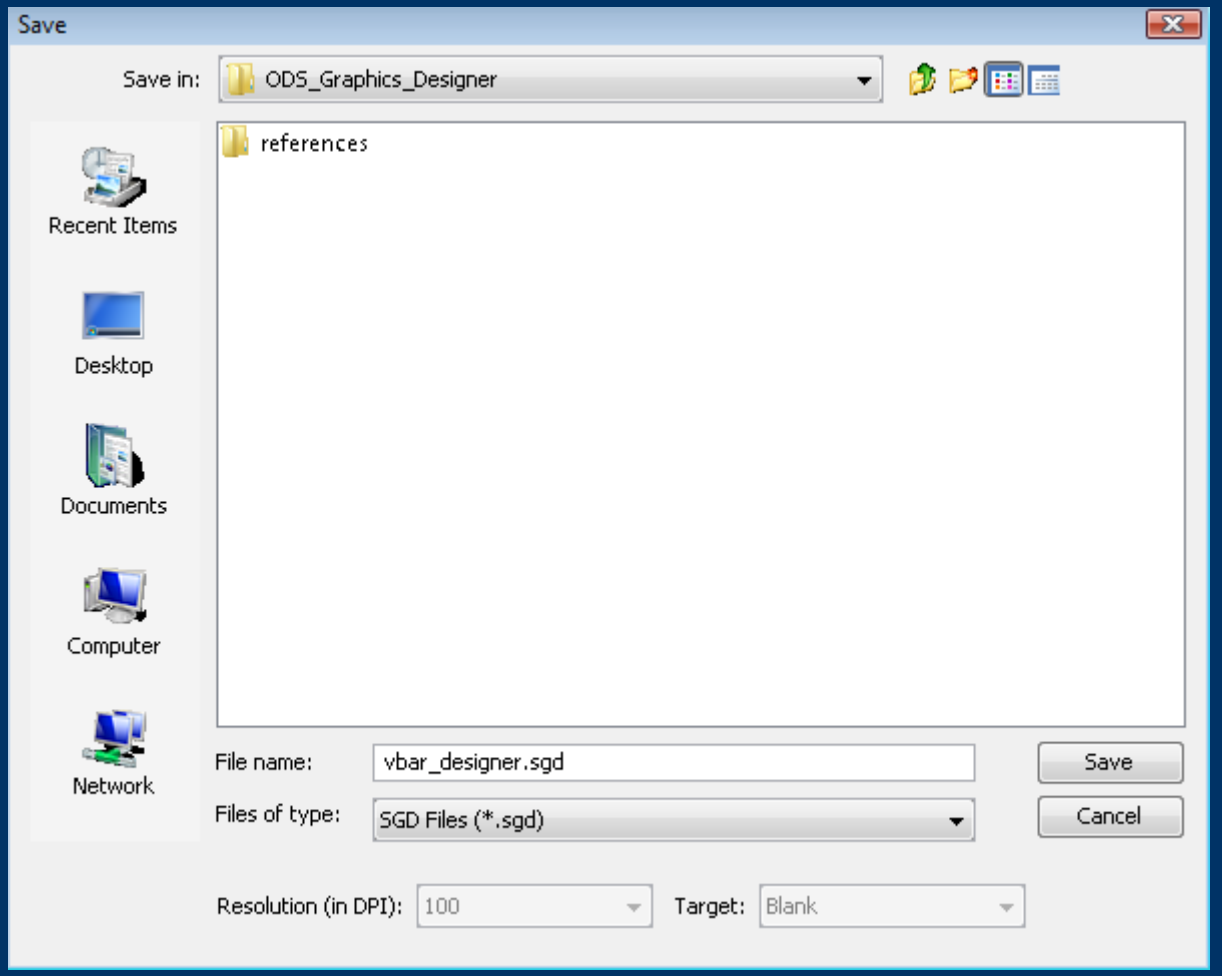
Code - Graph

```
proc template;
define statgraph sgdesign;
dynamic _AGE _AGE2 _HEIGHT;
begingraph;
entrytitle _id='title2' halign=center 'Panel title';
entryfootnote _id='footnote' halign=left 'Panel footnote ';
layout lattice _id='lattice' / columndatarange=data columngutter=10 rowdatarange=data
rowgutter=10 rows=2;
      layout overlay _id='overlay' / xaxisopts=(display=(LINE ));
        barchart _id='bar' x=_AGE / name='bar' stat=freq;
      endlayout;
      layout overlay _id='overlay2';
        scatterplot _id='scatter' x=_AGE2 y=_HEIGHT / name='scatter';
      endlayout;
endlayout;
endgraph;
end;
run;
```

Saving the template as a Designer file (*.sgd)

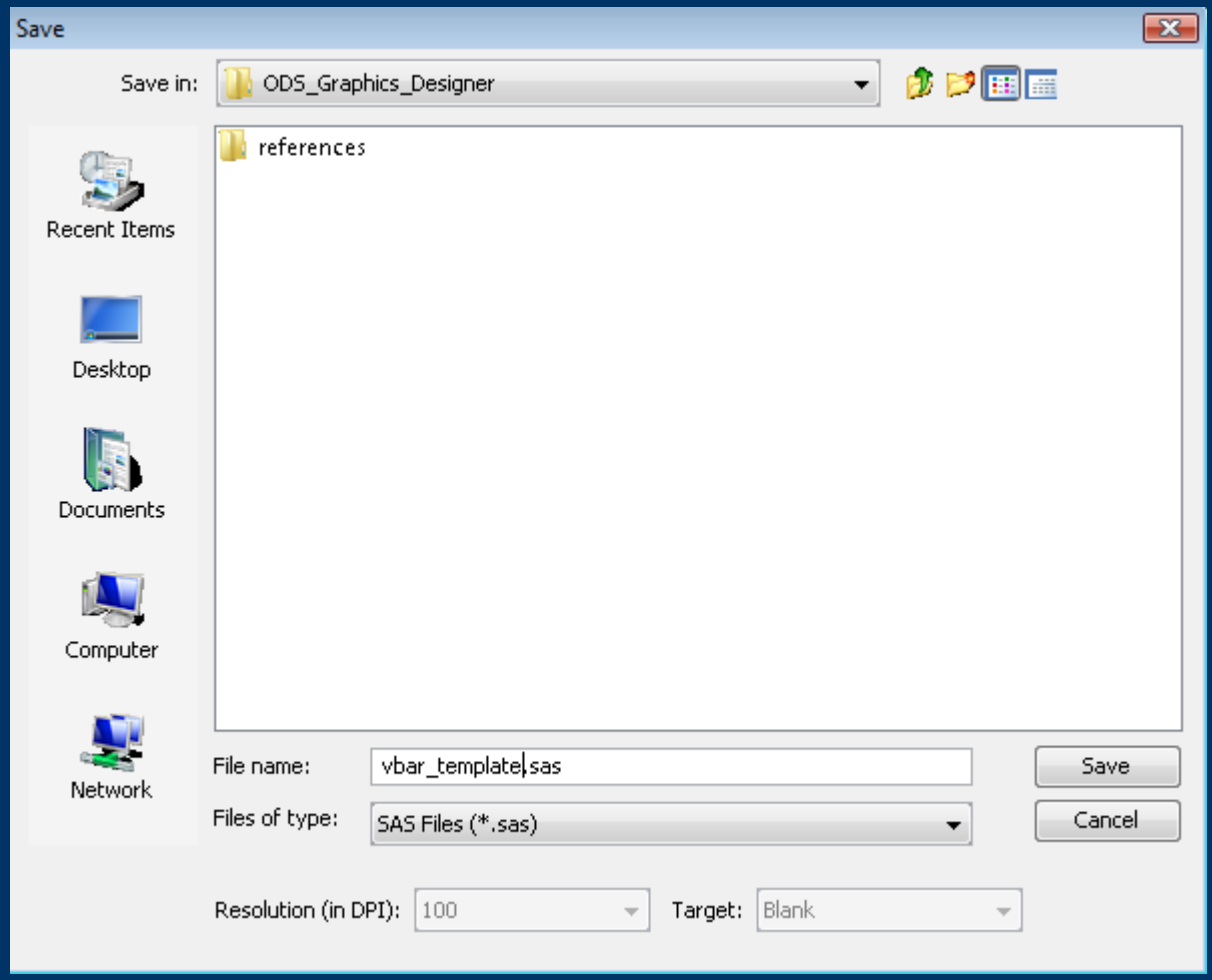
If you click on the Graph Window, then click the File menu option, the template can be saved as a Designer File by clicking Save As...

The template can also be saved to the Gallery by clicking Save to Gallery.



Saving the template as a SAS program (*.sas)

If you click on the Code Window, then click the File menu option, the template can be saved as a SAS Program by clicking Save As...



Summary and conclusions

- As a Java 1.5 application it will run on Windows, UNIX and Linux, but the application is sensitive to the Java release installed, so it is currently easier to use on Windows.
- This is a pre-production version, but is rumoured to be included as a production version in SAS 9.3.
- Templates generated are all called “sgdesign” by default, so will need to be edited before including in production applications. However, the SAS code generated is very robust.
- This is a good starting point to learn about and experiment with ODS Graphics templates.

Contact details

Philip R Holland, Holland Numerics Ltd
94 Green Drift, Royston, Herts SG8 5BT, UK
phil@hollandnumerics.com
www.hollandnumerics.com

This paper and presentation can be downloaded from:
www.hollandnumerics.com/SASPAPER.HTM