

VIEWS

International SAS Programmer Community



News, Hints, Tips and Information for SAS® Users

Issue 29, 1st Quarter 2005

New Year brings a new look for *VIEWS*. The *VIEWS* Committee decided to acknowledge our growing international membership, and has renamed *VIEWS* as the "International SAS Programmer Community". For more details about the changes to *VIEWS* see the open letter to members from the *VIEWS* Committee below. The aim of *VIEWS News* remains, as always, to keep SAS programmers informed of new techniques and developments. In keeping with our 'International' title, I would like to welcome 2 new *VIEWS* consultants, John Adams, from the US, and Peter Crawford from the UK, as well as articles from our regular contributors on SAS/GRAPH and SAS Metadata.

If you would like to contribute an article, or discuss the possibility of doing so, please send an email to me at newsletter@views-uk.org. A list of possible subjects for your articles can be found on the *VIEWS* web site, and anyone is very welcome to add to that list by sending emails with your own questions to newsletter@views-uk.org.

Philip R Holland (Newsletter Editor)

Membership of *VIEWS* is **free** and you can register for notification of all future *VIEWS* events by emailing us at membership@views-uk.org, making certain you include your name and postal address. If you would like to receive email notification whenever a new issue of the free quarterly *VIEWS News* is published on the web site, please remember to also include your email address.

News

The Future of *VIEWS*

VIEWS is changing! Our new name is "*VIEWS* International SAS Programmer Community" which we feel reflects our activities and purpose better.

We are not planning to run a conference in 2005, and in its place we would like to draw your attention to the SAS User Forum UK. We feel that the revitalised SAS Forum UK provides for many of the activities that were provided by the *VIEWS* conference, and would like to take this opportunity to encourage all *VIEWS* members to attend the SAS Forum UK, which will be held in November 2005. We are currently talking with SAS about the possibility of *VIEWS* moderating some Special Interest Group Sessions at the SAS Forum UK, where we can provide a further technical medium for networking and interchange between SAS Professionals. Provisional titles for these have been suggested as "Tips and Techniques SIG" (Coders' Corner) and "SAS Professionals SIG" (SAS Consultants and Users).

The *VIEWS* International SAS Programmer Community web site at www.views-uk.org will continue, along with the regular *VIEWS* newsletter. The web site and *VIEWS News* will be updated over the next few months to incorporate the new name and focus.

Thank you for your continuing support for *VIEWS*.

The *VIEWS* Committee

Ask The Consultant

This part of VIEWS News is where you can get your technical questions answered. Send your questions to the Editor.

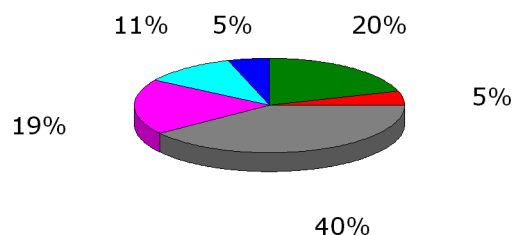
3D versus 2D Presentation Graphics

Q: What are appropriate SAS/GRAPH applications of 3D for two variables?

A: [SAS 8] Recently I found a way to use 3D graphics for two variables without detracting from effective and efficient graphic communication, but first I must demonstrate one type of 3D chart that I still cannot endorse, regardless of its popularity.

For Figure 1, consider the fact that the pie chart is a visual aid used expressly to facilitate comparison of shares of the whole:

Figure 1. 3D Pie Chart
Distorts the Apparent Relative Size of Slices



11% slice is almost as wide as 19% slice
5% slice at rear is twice as big as 5% slice at the right

Here is the code used to create Figure 1:

```
DATA work.ViewsFeb2005;  
  Category = 'A'; Value = 10; OUTPUT;  
  Category = 'B'; Value = 40; OUTPUT;  
  Category = 'C'; Value = 10; OUTPUT;  
  Category = 'D'; Value = 22; OUTPUT;  
  Category = 'E'; Value = 38; OUTPUT;  
  Category = 'F'; Value = 80; OUTPUT;  
RUN;
```

```

GOPTIONS RESET = ALL;
GOPTIONS BORDER;
GOPTIONS DEVICE = PNG GSFNAME = anyone;
GOPTIONS VPOS = 20 YMAX = 3.0 IN
          VSIZE = 3.0 IN YPIXELS = 900;
GOPTIONS HPOS = 34 XMAX = 3.25 IN
          HSIZE = 3.25 IN XPIXELS = 975;
GOPTIONS HTEXT = 1.0 FTEXT = 'Verdana'
          CTEXT = 'CX000000';

TITLE2 H = 1 F = 'Georgia' C = CX000000 J = L
'   Figure 1. ' C = CXFF0000 '3D Pie Chart'
J = L '   Distorts the Apparent'
'   Relative Size of Slices';
FOOTNOTE1 H = 1 F = 'Georgia' C = CXFF0000
J = L
'   5% slice at rear is twice as big'
'   as 5% slice at the right'
J = L
'   11% slice is almost as wide'
'   as 19% slice';
FOOTNOTE2 H = 0.5 F = NONE ' ';

FILENAME anyone "C:\3DPieChart.png";

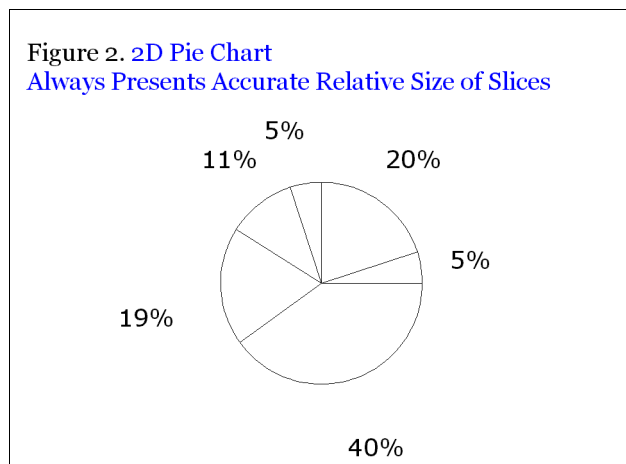
PROC GCHART DATA = work.ViewsFeb2005;
  PIE3D Category / SUMVAR = Value NOHEADING
        SLICE = NONE VALUE = NONE
        PERCENT = OUTSIDE;

RUN;
QUIT;

FILENAME anyone CLEAR;

```

Figure 2 does the job correctly:



Besides changing the **TITLE** text, the code changes used to create Figure 2 are:

```

GOPTIONS VPOS = 16 YMAX = 2.4 IN
          VSIZE = 2.4 IN YPIXELS = 720;
PATTERN1 V=PEMPTY;

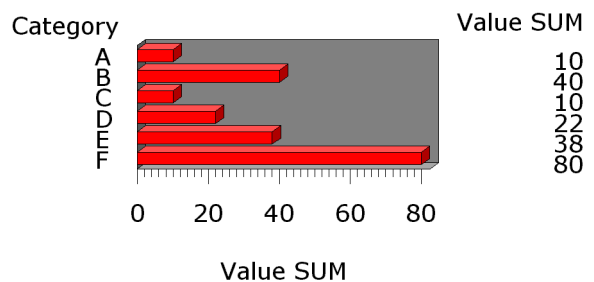
PIE Category / SUMVAR = Value NOHEADING
        SLICE = NONE VALUE = NONE
        PERCENT = OUTSIDE
        COUTLINE = CX000000;

```

Without **COUTLINE=CX000000**, the empty pie slices would be outlined with six different colours.

Believing in the power of simplicity, I cannot recommend the default bar chart in shown in Figure 3:

Figure 3. Default 3D Bar Chart
Needless Complexity



Here is the code used to create Figure 3:

```

GOPTIONS RESET = ALL;
GOPTIONS BORDER;
GOPTIONS DEVICE = PNG GSFNAME = anyone;
GOPTIONS VPOS = 16 YMAX = 2.4 IN
          VSIZE = 2.4 IN YPIXELS = 720;
GOPTIONS HPOS = 34 XMAX = 3.25 IN
          HSIZE = 3.25 IN XPIXELS = 975;
GOPTIONS HTEXT = 1.0 FTEXT = 'Verdana'
          CTEXT = 'CX000000';

TITLE2 H = 1.0 F = 'Georgia' C = CX000000 J = L
'   Figure 3. ' C = CXFF0000
'Default 3D Bar Chart'
J = L '   Needless Complexity';

FILENAME anyone "C:\Default3DBarChart.png";

PROC GCHART DATA = work.ViewsFeb2005;
  HBAR3D Category / SUMVAR = Value;

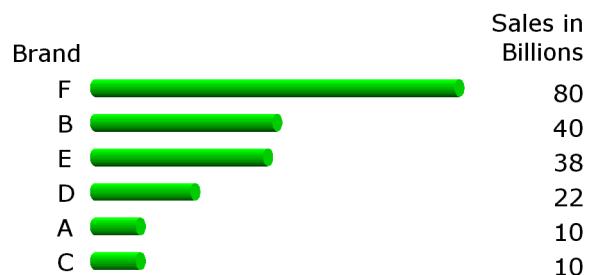
RUN;
QUIT;

FILENAME anyone CLEAR;

```

Figure 4 is the only kind of 3D chart for two variables that I can recommend:

Figure 4. Optimized 3D Bar Chart
Interest Added, But Still Communication-Effective



Besides changing the **TITLE** text, the code changes used to create Figure 4 are:

```
PATTERN1 C = CX00FF00;
AXIS1 LABEL = ('Brand') MAJOR = NONE
      MINOR = NONE STYLE = 0 NOPLANE;
AXIS2 LABEL = NONE MAJOR = NONE MINOR = NONE
      STYLE = 0 NOPLANE VALUE = NONE;

HBAR3D Category / SUMVAR = Value NOFRAME
      SHAPE = CYLINDER
      SUMLABEL = 'Sales in Billions'
      WIDTH = 0.6 SPACE = 0.6
      MAXIS = axis1 RAXIS = axis2
      DESCENDING;
```

This **HBAR3D** statement includes the **NOFRAME** option to suppress the unneeded “back plane” that appeared in Figure 3. Adding the **NOPLANE** options to the **AXIS** statements suppresses the unneeded “axis planes”. The alternative shapes of **BLOCK**, **HEXAGON**, **PRISM**, and **STAR** would add no communication value. Also, when using those, to override the default colour (using a **PATTERN** statement, as above) it is necessary to also assign a contrasting colour **COUTLINE** value in the **HBAR3D** statement, in order to see the structure of the expected shape.

LeRoy Bessler, Assurant Health

Editors for SAS Coding

An interview with Peter Crawford, Senior SAS Consultant at Crawford Software Consultancy Ltd:

Editor: We all have our favourite editors that we use for writing SAS code. Do you have a preferred editor?

Peter Crawford: At the next client site anything is possible, but I haven't yet enjoyed one where I'm free to choose anything I prefer, over those available. The choice is platform specific, or the alternative is SAS text editor. At least with the SAS text editors I can carry the learning forward from client site to client site. Fortunately, the steamroller of Microsoft Office ensures most SAS clients run on Windows. For those who continue to use zOS, OS390 or MVS, I return to my "SAS-childhood". It was amazing how many eyes were opened with the capabilities of their chosen platform - beyond the "out-of-the-box" defaults. I could make the traditional "keyhole" window sparkle with double the number of lines of code, and with the syntax all highlighted. Unix offered many neat features of course, but now I'm taking off on Windows...

Editor: Are there any features of the SAS text editors that exist on all platforms, but you can't find in other editors? In other words, what are the really useful features?

Peter Crawford: There are many features I find really useful with SAS editors, but simply the most important to me, is that SAS editors will be available, and familiar in the first few days at a new client. I won't be spending time to acquaint myself with a new editor. Setting "availability" aside, top, comes integration with SAS - for dictionary tables interacting with **%MACRO**. Dictionary tables provide a whole range of information relevant to the programs I'm editing. The editing environment can be extended with syntax (macro) that is in the middle of my expertise. Extending any other editor requires syntax in "yet another" language that isn't SAS. I can create commands that call SAS macros which use things like **dictionary.columns** to work with variable lists. These custom commands get saved among

spare "function" keys in my preferred environment. But I'm always emphasising that instead, they can be called via a toolbar icon.

Here is the simplest command extension, to insert the current timestamp after the next blank character, which can be used in all SAS text editors:

```
C ' ' "%now(fmt=DATETIME14) "
```

where the **%now** macro is:

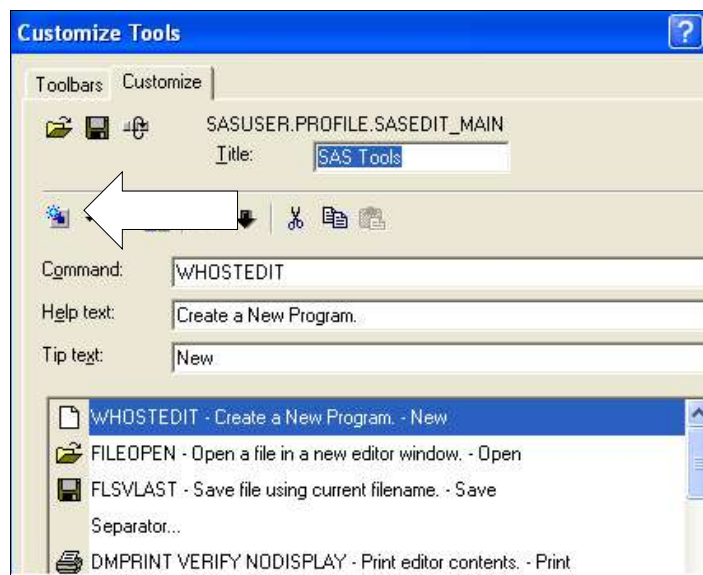
```
%MACRO now(fmt=DATETIME23.3)
  %SYSFUNC(DATETIME(), &fmt)
%MEND now;
```

Editor: Can you demonstrate how to incorporate such a command as a toolbar icon?

Peter Crawford: When the focus is in the Enhanced Editor, select Tools from the main SAS menu, and Customize from the Tools sub-menu.



Click on the "Customize" tab (as marked above). That opens the "Customize Tools" form, where you control the toolbar and the behaviour of its items. Now add a new tool to the tool bar:

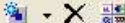


Customize Tools

Toolbars: Customize

 SASUSER.PROFILE.SASEDIT_MAIN

Title:

Command:

Help text:

Tip text:

 WHOSTEDIT - Create a New Program. - New

 FILEOPEN - Open a file in a new editor window. - Open

 FILELAST - Open the last file in the current session. - Open

Customize Tools

Toolbars | **Customize**

SASUSER.PROFILE.SASEDIT_MAIN

Title: SAS Tools

Command: c'''%now(fmt=datetime13)'''

Help text: insert timestamp at the next blank

Tip text: timestamp

c'''%now(fmt=datetime13)' - insert timestamp at the next blank - tim

WHOSEDIT - Create a New Program. - New

mySAS

File Edit View Tools Run Solutions Window Help

Log - timestamp

NOTE: SAS initialization used:
 real time 1.19 seconds
 cpu time 0.91 seconds

Editor - Untitled2 *

1

Output - (Un... Log - (Unitt... Editor - Unti... Editor - Unt...

insert timestamp at the next blank C:\Documents and Settings\Peter Crawford Ln 1, Col 1

Editor: Thank you.

If you have a useful hint or tip, please send it to the Editor and share it with the VIEWS membership.

[SAS 8] Here's an undocumented feature of SAS that may be useful to programmers that create many temporary datasets during the processing of their programs and don't want users to look at them:

Name all the temporary datasets with a prefix of "**to**" (e.g. **work_totest**) and they will not show up in the Explorer window. However, you can access them like any other dataset. Of course, this should not be a substitute for proper cleanup at the end of a program (e.g. deleting temporary datasets, etc.).

John Adams, Boehringer Ingelheim

Exploiting New Metadata with the Data Step Interface

[SAS 9] SAS 9 metadata is a rich source of information, with powerful XML and Java-based interfaces that are fully programmable and very flexible. But what if you don't have a Java programmer handy? Or the time to learn it and the necessary XML yourself? Is there no other way to read and use this information?

All is not lost, SAS have included a data step interface! Although the documentation of this is not easy to find unless you know where to look (see references at the end of this article). A whole set of functions are available, **METADATA_GET...**, **METADATA_SET...** and so on allow you to both query and update, or even create, all kinds of SAS 9 metadata. The functions will even work in SCL programs, although this is undocumented as far as I can see.

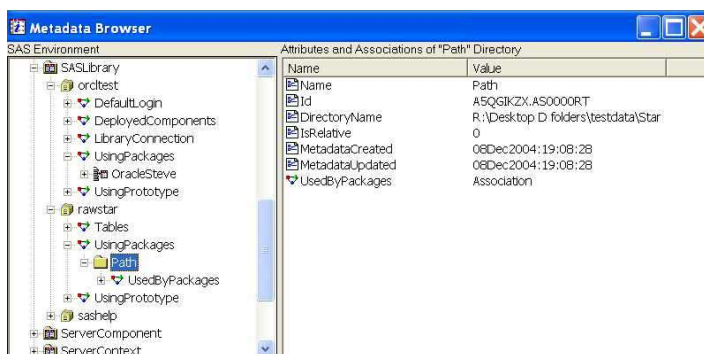
I should mention that I would not advocate this as a general alternative to Java interfacing. It is very long-winded to create objects, for example, but it does have many practical uses such as simple metadata content reporting.

To illustrate what is possible, this article describes creating a small 'utility' step to report all the SAS Data Libraries defined in a SAS 9 Repository, along with their directory path details.

First some basics about the SAS 9 metadata model; this is strongly object-oriented, based on over 150 metadata types. The metadata model is similar in some ways to the older SAS/Warehouse Administrator (WA) metadata model, but it now spans everything to do with the SAS environment, from ETL through BI reporting, data mining, portal delivery and so on; it is therefore hosted by a metadata server that all SAS applications can use.

The other notable difference comes from how information is structured, using Attributes, Properties and Associations. But whereas the older WA metadata used attributes for many useful items, the new metadata stores only basic information about the object in Attributes; nearly everything important is a result of Association between one object and another.

To understand the Attributes and Associations in use, I recommend using the Metadata Browser in Base SAS 9 (command *metabrowse* will open the dialog) to help you understand the documentation. Open this on a metadata repository that you know the contents of and explore it before writing any programs to do the same (see below).



The particular Association I will exploit here is between SASLibrary objects, used to define libnames, and Directory objects that define the physical path used. Note that non-SAS libnames such as RDBMS's use a different Association (to a DatabaseSchema).

```
/* Simple data step reporting of
   SAS 9 Library Metadata */

DATA _NULL_;
  LENGTH name uri_lib uri_path path $256;
  * iterate until no matching objects;
  DO UNTIL (lrc < 0);
    * get the next SASLibrary definition
      with non-blank name;
    n + 1;
    lrc = METADATA_GETNOBJ(
      "omsobj:SASLibrary?@Name ? '",
      n, uri_lib); *1*;
    IF lrc > 0 THEN DO;
      * get attribute and association details;
      rc = METADATA_GETATTR(
        uri_lib, 'Name', name); *2*;
      rc = METADATA_GETNASN(
```

```
        uri_lib, 'UsingPackages', 1,
        uri_path);
      rc = METADATA_GETATTR(
        uri_path, 'DirectoryName',
        path); *3*;
      PUT / 'Library name= ' name / +8 path=;
    END;
  END;
  STOP;
RUN;
```

Note: in order to run this code you will also need an options statement (not shown) defining metaserver, metaport, metauser, metapass and metarepository.

To locate each SASLibrary in turn I use a call to **METADATA_GETNOBJ**, using a generalised search argument as the input universal resource identifier (**uri**); this returns the specific **uri** for the *n*th SASLibrary (see *1* above).

Then for each one I call **METADATA_GETATTR** for a specified Attribute, the 'Name' (see *2* above). Finally the directory path is returned using two calls: **METADATA_GETNASN** to locate the Association to some Directory information and another **METADATA_GETATTR** to read the DirectoryName Attribute of the Directory object (see *3* above).

Output from this will be something like that shown below: notice that the Oracle library, **orctest**, does not report path information; it would require slightly different code to return the Schema details, but hopefully you can see the concept here.

```
Library name= orctest
  path=

Library name= rawstar
  path=R:\Desktop D folders\testdata\Star

Library name= sashelp
  path=

NOTE: DATA statement used (Total process time):
      real time           0.09 seconds
      cpu time            0.01 seconds
```

Documentation: These data step functions do not appear in Base SAS sections of the SAS 9 documentation. Instead the **SAS Open Metadata Interface: Reference** provides the important material. But beware, in SAS 9.1.3 only the OnlineDoc version of this has the detailed information on the data step functions, as the PDF version just lists them.

Steve Morton, Applied System Knowledge

Formats, Options, and Functions

This section is devoted to the description of useful, or unusual, SAS Formats, Options, and Functions.

[SAS 9] At last, there are some new SAS Formats that can display currency values in Pounds Sterling:

```
DATA _NULL_;
  x = 1234.56;
  PUT x= NLNMLGBP. x= NLNMLGBP.;
RUN;
```

produces:

x=£1,234.56 x=GBP1,234.56

The **NEWS=external_file** configuration or invocation option is a System Option that is not widely used, but has been available since the earliest versions of the SAS System. It includes the contents of the specified text file in the SAS Log immediately after the copyright, version, licence and platform Notes. SAS Administrators can use this text file to pass on SAS-related news to SAS users via the SAS Log.

[SAS 9] The **ROUND()** function has been available in SAS for a long time, but it doesn't always work the way you want it to. The new **ROUNDE()** function rounds the first argument to the nearest multiple of the second argument, and returns an even multiple when the first argument is halfway between the two nearest multiples.

In Brief

- Back issues of *VIEWS News* are available from the *VIEWS* web site.
- Phil Mason now has a daily SAS tips newsletter, which can be requested by going to www.woodstreet.org.uk and then clicking on Tips. Previous tips can also be found on the web site.
- SAS users looking for contracts, jobs, services and products may find what they are looking for on the Sponsored Links web page at www.hollandnumerics.com/careers.

Diary

Are you organising an event that would be of interest to the VIEWS readership? Let us know as we are interested in all non-profit making events related to SAS.

March 2005

1 Changes and Enhancements in SAS Version 9 for the Applied Statistician, Amadeus Software -FREE!

See the Amadeus Software web site, www.amadeus.co.uk, for further details.

April 2005

5 Changes and Enhancements in SAS Version 9 for the Applied Statistician, Amadeus Software -FREE!

See the Amadeus Software web site, www.amadeus.co.uk, for further details.

10-13 **SUGI**, Philadelphia, USA

The SAS User Group International (SUGI) Conference is held each year and focuses on the technical aspects of SAS software.

June 2005

16 Changes and Enhancements in SAS Version 9, Amadeus Software -FREE!

See the Amadeus Software web site, www.amadeus.co.uk, for further details.

21-23 **SAS Forum International**, Lisbon, Portugal

The SAS Forum International is held each year and focuses on the more commercial aspects of SAS software.

September 2005

20 Data Analytics and Reporting, Amadeus Software -FREE!

See the Amadeus Software web site, www.amadeus.co.uk, for further details.

November 2005

tba SAS Forum UK

The SAS Forum UK is held each year and presents the latest SAS developments to SAS users in the UK. See www.sas.com/uk for the latest news.

15 Tricks of the Trade, Amadeus Software -FREE!

See the Amadeus Software web site, www.amadeus.co.uk, for further details.

All VIEWS event information will be posted and/or emailed to registered members of VIEWS, and will include an event application form. Please send your queries about any VIEWS events to event@views-uk.org, and don't forget to look at the web site for the latest news.

Contacts

John Adams
Boehringer Ingelheim,
Ridgefield, Ct, USA

Email:
jadams@boehringer-ingelheim.com

LeRoy Bessler
Assurant Health,
Milwaukee, USA

Email:
bessler@execpc.com

Peter Crawford
Crawford Software
Consultancy Ltd

Email:
cscl@blueyonder.co.uk
Tel: +44-(0)7802-732254

Steve Morton
Applied System
Knowledge Ltd

Email:
steve.morton@appliedsystem.co.uk
Tel: +44-(0)1491-411977
Web: www.appliedsystem.co.uk

SAS UK

Tel: +44-(0)1628-486933
Web: www.sas.com/uk

VIEWS

Web Site

www.views-uk.org

Chairman

Email:
chair@views-uk.org

Philip R Holland
Newsletter Editor

Email:
newsletter@views-uk.org
Tel: +44-(0)7714-279085
Fax: +44-(0)1763-244497
Web: www.hollandnumerics.co.uk

Membership secretary

Email:
membership@views-uk.org

Events secretary

Email:
event@views-uk.org

***VIEWS News** is published quarterly by VIEWS International SAS Programmer Community. VIEWS is a not-for-profit organisation. Our Mission Statement and Rules can be viewed on our web site together with a list of current committee members.*