

VIEWS

International SAS Programmer Community



News, Hints, Tips and Information for SAS® Users

Issue 71, 1st Quarter 2023

Even though Foundation SAS is being pushed to one side by SAS Viya, this newsletter continues to bring you information about making the most of SAS 9. This newsletter issue includes another useful SAS programming tip from Nina L Werner, I asked ChatGPT a question about SAS Studio and WPS (now called Altair SLC), and Richard Carson discusses the configuring of SAS processes without editing them in the first part of a series of articles about KT. As usual I have publicised some SAS-related events for the next few months, and I have also added some more SAS formats, options and functions that you may or may not have come across before, but you might find interesting, and even useful.

If you would like to contribute an article, to re-visit and improve an existing article, or just discuss the possibility of doing so, please feel free to send an email to me at views-uk@hollandnumerics.org.uk.

Philip R Holland (Newsletter Editor)

Membership of VIEWS is **free** and you can register for notification of all future VIEWS News issues by registering at blog.hollandnumerics.org.uk, and then emailing views-uk@hollandnumerics.org.uk, or using the blog site menu, to request access to the VIEWS UK forum. Members will receive an email notification whenever a new issue of the free quarterly VIEWS News is published on the web site, please inform us when you change your email address, otherwise your registration may be cancelled.

Did You Know?

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

KT: Part 1. Configuring a recurring SAS process without editing code

What this is about

This design is intended for processes that require repeatable execution, such as datamart loads, credit scoring, and regular monitoring or reporting. It has the following objectives:

- Creating a “universal” code base for running regular or scheduled processes that can be reconfigured, when required, without editing core SAS statements (but see the caveat below).
- Using text configuration files in a simple, readily edited format.
- Separation of the core code base from configuration inputs, data and outputs.

- With the core code base isolated from tasks and data, a single instance can be used to run different projects without the need for replication, providing high reliability for the process.
- Ability to be extended by adding new tasks, and exchanging or removing existing tasks, solely by updating configuration inputs.
- Ability to run without modification on different operating systems (Windows, Linux, UNIX and, hopefully, MVS) in batch or interactive mode.
- Run as stand-alone SAS code or from within Enterprise Guide or SAS Studio (or even DI Studio, if required, though not taking advantage of its unique features).
- Choice of running at entry level or in production mode with compiled macros.

What it is not about

This design is not relevant for one-off or iterative processes such as exploratory data analysis, statistical analyses or model building.

Nevertheless, during model building, it could be useful to build a repository for analysis, to be re-purposed as a basis for reporting trends or monitoring model accuracy.

Caveat

Of course no fixed code base can be configured to run arbitrary tasks. The core KT code is not concerned with data manipulation, which is all done at the individual task level. For convenience, the KT code base can be extended to include commonly used tasks, but individual projects with unique requirements can define their own tasks to be used in conjunction with KT.

Design

This design envisages separation of a typical process into five components:

- External data.
- Individual tasks used to import data and restructure it to meet process requirements.
- External outputs.
- Configuration inputs.
- KT, the core process execution code including a toolbox of some commonly used task operations.

The KT core process consists of 3 nodes:

- nodeStem: the entry point for the process, handles housekeeping tasks such as loading the core and project configuration data and managing process macros, formats and options.
- nodeBranch: which determines which tasks are run and their order.
- nodeConnect: which joins nodeBranch to individual tasks and provides the parameters for task execution.

For each project to run on the KT core just 2 pieces of information need to be provided at initiation; the location of the nodeStem code and the location of the individual project task configuration files.

Configuration file format

KT will read configuration files into tables in SAS control or configuration libraries, so the format used for the text configuration files can be specified in a settings configuration. As the task of importing configuration data is a discrete macro within nodeStem, it could be swapped out for an alternate version to interpret configuration data in another format. Configuration data in the following articles will be presented in tabular form, with a code snippet for importing PDF tabular data. For brevity and simplicity the project design here assumes CSV files for configuration. The initial version of this project used INI files (https://en.wikipedia.org/wiki/INI_file), which will be an option available in the full implementation. XML, YAML or JSON inputs will require additional code.

KT will expect to have configuration tables with the following character columns (column widths can vary):

- Section: a grouping variable (optional).
- Parameter: the key, name of a quantity to be used in the process.
- Value: the property associated with the parameter (optional). Numeric values will be read as text.
- Comment: an optional string for clarification of purpose.

Configuration tables will be stored in a control library for KT core settings, and in config libraries for individual projects. For simplicity configuration tables will not be concatenated.

Invocation

The KT core and toolbox code files will eventually be available from a GIT repository. Individual projects using KT should have their own distinct folder structure to hold imported data and exported results, along with configuration files and any bespoke task code.

A project based on KT core code typically would be invoked from the project configuration folder, which would initially link to the KT code repository.

KT Demonstration

I am using the data repository for the 2019 Novel Coronavirus Visual Dashboard operated by the Johns Hopkins University Center for Systems Science and Engineering (JHU CSSE), in particular the global confirmed cases data on GitHub CSSEGISandData [1]. This series is well curated (within the limits of the supplied source data) and of topical interest. It does, however, pose some challenges, because the time series data for each country is contained in a single row. This is an unnecessary complication for my purpose, but a “narrow” version of the data is available from the (UN) Humanitarian Data Exchange [2].

Richard C Carson

[1] https://github.com/CSSEGISandData/COVID-19/blob/master/csse_covid_19_data/ (time series).

[2] <https://data.humdata.org/dataset/novel-coronavirus-2019-ncov-cases?> Select the “narrow” confirmed file.

Ask The Consultant

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

How to Self-Document a Compiled Macro

Q: How do you know what a compiled macro does and how to use it?

A: [SAS] Document your macro with HELP text.

When you write a macro program, help yourself and others who may use it later by offering a help message comment box. Here's an example.

```
%macro mymacro(help, parm=Y)
  /DES='mymacro does something for me';
  *STORE SOURCE;
  %if &help.=? or %upcase(&help.)=HELP %then %do;
    %put;
    %put %nrstr(=====);
    %put %nrstr(Macro call example:
    mymacro(sasuser, parm=Y));
    %put %nrstr(Get help by submitting
    mymacro(?); or mymacro(help); );
    %put;
    %put %nrstr(First parameter is library name or
    help, );
    %put %nrstr(Second parameter parm=Y (default)
    or N, Others );
    %put;
    %put %nrstr(Default macro call is:
    mymacro(parm=Y); for WORK library);
    %put;
    %put %nrstr(Other text explanation);
    %put %nrstr(Nina L. Werner 2/1/23);
    %put %nrstr(=====);
    %goto exit;
  %end;
  ** >>here is all the actual macro code<< **;
  %exit;
%mend mymacro;
```

Run the macro (help) to see the message in your SAS Log.

```
%mymacro(?);

=====
Macro call example: %mymacro(sasuser, parm=Y);
Get help by submitting %mymacro(?); or
%mymacro(help);

First parameter is library name or help,
Second parameter parm=Y (default) or N, Others

Default macro call is: %mymacro(parm=Y); for
WORK library

Other text explanation
Nina L. Werner 2/1/23
=====
```

Nina L Werner, Wisconsin

ChatGPT: Compare SAS Studio and World Programming Workbench software

SAS Studio and World Programming Workbench are both software programs designed for data analysis and management. However, there are some key differences between the two.

SAS Studio is a web-based interface that allows users to access the SAS analytics platform from any device with an internet connection. It offers a comprehensive suite of tools for data analysis and visualization, including SAS programming languages, a code editor, and a range of libraries and modules for data manipulation, statistics, and machine learning.

World Programming Workbench, on the other hand, is a software environment that emulates the SAS programming language, allowing users to write and run SAS code on their own computer. It is designed for businesses and organizations that do not have access to the SAS platform, but still need to analyse and process large amounts of data. World Programming Workbench includes a code editor, a workspace, and a range of libraries and modules for data analysis and management.

In conclusion, SAS Studio is ideal for businesses that need to access the SAS analytics platform from multiple devices, while World Programming Workbench is ideal for organizations that need to run SAS code on their own computer.

ChatGPT (with a prompt from Philip R Holland)

Note that ChatGPT is just a pattern-matching language engine that has been trained on a vast library of books and articles published up to 2021. It can paraphrase its training data, but can not invent new information!

Formats, Options, and Functions

This section is devoted to the description of useful, or unusual, SAS Formats, Options, and Functions. Please read the SAS documentation to find out more about them.

[SAS] The SAS formats **WORDDATEw.** and **WORDDATXw.** display SAS dates with full words for months. 01Mar2023 is displayed as “March 1, 2023” by **WORDDATE.** and as “1 March 2023” by **WORDDATX.**

[SAS 9.2] The **ATTRID=** and **GROUP=** options used on plot statements within **PROC SGPLOT**, **SGPANEL** and **SGSCATTER** must be used together with the **DATTRMAP=** options on the PROC statement to standardise the colours and patterns of text, lines and filler in plots.

[SAS 8.1] While the SAS holiday functions **HOLIDAY()**, **HOLIDAYCK()**, **HOLIDAYCOUNT()**, **HOLIDAYNAME()**, **HOLIDAYNX()**, **HOLIDAYNY()** and **HOLIDAYTEST()** only include holidays from the US and Canada, although some are still relevant elsewhere in the world, like Christmas, Boxing, NewYear, Easter and Valentine days. However, all is not lost in the UK, as most of the bank holidays without fixed dates do have a fixed day of the week within a month, so **NWKDOM()** can be used instead to find May Bank Holiday Monday, Spring Bank Holiday Monday and August Bank Holiday Monday.

In Brief

- Back issues of *VIEWS News* are available from the VIEWS UK forum at **blog.hollandnumerics.org.uk**. Please register for free membership on the blog, and then request free access to the VIEWS UK forum.
- The SAS Professional Forum group on LinkedIn, which currently has over 39,100 members. Everyone is welcome to request to join, but discussions here are restricted to SAS-related topics: **www.linkedin.com/groups?gid=70702**.
- The SAS Opportunities group on LinkedIn, which currently has over 16,900 members. Everyone is welcome to request to join, but discussions here are also restricted to SAS-related topics: **www.linkedin.com/groups?gid=805217**.
- In the UK and Ireland the SUGUKI group had been holding regular meetings in London, and also occasional meetings in Bristol, Dublin, Edinburgh, Leeds and Manchester. For more details and to register for future events please go to **meetup.com/SUGUKI**.
- Fancy joining the SAS EMEA Analytics Explorers, then please send your **country**, **name**, **email address** and **company** with your permission to pass these details on to SAS to **views-uk@hollandnumerics.org.uk**. Other regions include Asia Pacific, Canada, Latin America and US, but to join these communities based on where you live, you will now need to provide this information to a current member of these communities, instead of me.
- Do you use Microsoft Visual Studio Code Editor? You may be interested in a SAS-related extension called **SASjs**. You can develop SAS applications and run code on SAS Viya, SAS 9 EBI or SASjs Server! Just set up the target, and click the running man to execute your SAS code. The SASjs Server could be your local SAS installation. **SASjs** is an open source project, and can be obtained from the Visual Studio Code Marketplace.
- Altair Analytics Workbench Community Edition provides named users with free access to Altair SLC software (formerly WPS) including a SAS language compiler with coding and workflow tools on unlimited data volumes. A Community Edition licence key will be issued that will activate the software for up to 12 months. At the end of this term, users may be invited to renew their licence and/or to upgrade to the latest available full version of the software. Download the software from **www.altair.com/altair-slc**. No technical support service is offered with the Community Edition, but assistance can be requested using the Community forums at **community.altair.com/community**. The current General Access version of Altair SLC is 4.4.6.

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.

May 2023

8-10 **SAS Innovate**, Orlando, FL, USA

SAS Innovate is a conference dedicated to analytics and AI for executives and how to transform existing data into faster, more effective decisions in an era of digital transformation, economic uncertainty, consumer demands and fierce competition. More details at www.clocate.com/sas-innovate.

14-17 **PharmaSUG**, San Francisco, CA, USA

Join us for the premier event of the year for statistical programmers, statisticians, data managers, researchers, and others who analyze data in the life sciences - PharmaSUG 2023! Full details at www.pharmasug.org.

November 2023

5-8 **PhUSE EU Connect**, Birmingham, UK

PhUSE Connect events bring together our global Community to collaborate on hot topics, share perspectives and discuss different approaches to guiding the healthcare industry landscape, both now and in the future. Full details at www.phuse-events.org.

Contacts

Richard C Carson

Email: richard@arcol-analytics.com

Nina L Werner,
Wisconsin

Email: Nina.Werner@dot.wi.gov

SAS UK

Tel: +44-(0)1628-486933

Web: www.sas.com/uk

VIEWS

Web Site

Web: blog.hollandnumerics.org.uk

Newsletter Editor

Email:

views-uk@hollandnumerics.org.uk

***VIEWS News** is published quarterly for the VIEWS International SAS Programmer Community by Holland Numerics Limited.*