Share your code with SAS® Packages
a Hands-on-Workshop

Bartosz Jabłoński - yabwon / Warsaw University of Technology

ABSTRACT

When working with SAS®code, especially when it becomes more and more complex, there is a point in time when a developer decides to break it into small pieces. The developer creates separate files for macros, formats/informats, and for functions or data too. Eventually the code is ready and tested and sooner or later you will want to share code with another SAS programmer.

Maybe a friend has written a bunch of cool macros that will help you get your work done faster. Or maybe you have written a pack of functions that would be useful to your friend.

There is a chance you have developed a program using local PC SAS, and you want to deploy it to a server, perhaps with a different OS. If your code is complex (with dependencies such as multiple macros, formats, datasets, etc.), it can be difficult to share. Often when you try to share code, the receiver will quickly encounter an error because of a missing helper macro, missing format, or whatever...

Small challenge, isn't it?

How nice it would be to have it all (i.e. the code and its structure) wrapped up in a single file - a **SAS package** - which could be copied and deployed, independent from OS, with a one-liner like: `%loadPackage(MyPackage)`.

In the presentation an idea of how to create such a **SAS package** in a fast and convenient way, using the **SAS Packages Framework**, will be shared.

We will discuss:
    (1) concept of how to build a package,
    (2) the tools required to do so (the framework),
    (3) overview of the process, and
    (4) how to build a package (with examples).

The intended audience for the presentation is intermediate or advanced SAS developers (i.e. with good knowledge of Base SAS and practice in macro programming) who want to learn how to share their code with others.

TECHNICAL REQUIREMENTS

The workshop assumes that you have available the following resources:
- a computer with internet access,

- a SAS session (may be local or remote), the interface can be: `SAS DMS`, `SAS EG`, or `SAS Studio`, can be `SAS 9.4` or `Viya` with `SPRE`
- the SAS session has:
  - direct internet access or
  - you are able to upload files to the computer hosting your session,
  - (nice to have but not required) the SAS session has `XCMD` option enabled.

Additionally, we assume that *you* wrote at least a few of those: a macro, a function (fcmp, proto, or casl), a format or informat, or an IML module. The more the better.

LINK TO WORKSHOP MATERIALS

Visit the link bellow to find all workshop materials.

$$\texttt{https://github.com/yabwon/HoW-SASPackages}$$

REFERENCES

[Carpenter 2012] Art Carpenter, "Carpenter's Guide to Innovative SAS Techniques", SAS Press, 2012

[Jablonski 2020] Bartosz Jabłoński, "SAS Packages: The Way to Share (a How To)",
    SAS Global Forum 2020 Proceedings, 4725-2020
    `https://www.sas.com/content/dam/SAS/support/en/sas-global-forum-proceedings/2020/4725-2020.pdf`
    extended version available at: `https://github.com/yabwon/SAS_PACKAGES/blob/main/SPF/Documentation`

[Jablonski 2021] Bartosz Jabłoński, "My First SAS Package - a How To",
    SAS Global Forum 2021 Proceedings, 1079-2021
    `https://communities.sas.com/kntur85557/attachments/kntur85557/proceedings-2021/59/1/Paper_1079-2021.pdf`
    also available at: `https://github.com/yabwon/SAS_PACKAGES/tree/main/SPF/Documentation/Paper_1079-2021`

ACKNOWLEDGMENTS

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at one of the following e-mail addresses:

$$\texttt{yabwon} \boxtimes \texttt{gmail.com} \quad \text{or} \quad \texttt{bartosz.jablonski} \boxtimes \texttt{pw.edu.pl}$$

or via the following LinkedIn profile: `www.linkedin.com/in/yabwon` or at the `communities.sas.com` by mentioning `@yabwon`.