How to HOW: Hands-on-Workshops Made Easy

Chuck Kincaid, Experis Business Analytics, Kalamazoo, MI

ABSTRACT

Have you ever attended a Hands-on-Workshop and found it useful? Many people do! Being able to actually try out the things that you're learning is a wonderful way to learn. It's also a great way to teach. You can actually see the people apply what they're learning.

Have you ever thought that it would be fun to teach other people in a hands on format? Maybe you weren't sure what it takes or how to approach the course. This presentation will help you with those questions and struggles. What to teach? How much to teach? How should I teach it? How is a Hands-on-Workshop different than lecture style? How much to put into PowerPoints? What if they ask me something I don't know? What if they have a computer problem?

All those questions that you have will be answered in this presentation.

WHAT IS A HOW?

People come to conferences for many reasons: the presentations, the special events, the challenges, the vacation before or after, the food, to find jobs, the demos, have time away from kids, network with other users, the tchotchkes, and on and on. The big reasons can be summarized as learning, networking and fun. There is great learning in the presentations and their papers. Users give great talks on a wide variety of facets of the SAS® System. If these were all we had, then the conferences would be well worth the time and money. But we have an even greater opportunity to get hands on experience with SAS through the Hands on Workshops, or HOWs.

From the earliest conference years, these Hands on Workshops have given people direct experiential learning with an aspect of SAS. Because the attendee is actually coding and clicking in these sessions they learn more and retain it longer.

The Hands on Workshop (HOW) is an opportunity for attendees to learn by doing on some topic. A presenter teaches them about that topic like in other presentations. In this case, the attendee, though, is following along with the instructor on a computer. They are doing exercises that give them hands on experience for the lesson just taught.

The HOWs are in their own track and usually about 1 hour and 20 minutes each. They can be about anything related to SAS that can be done with the software installed on the computer and usually without the internet. Over the years, some of the topics have been

Merge Processing in SAS Software	Moving Data and Analytical Results between SAS and Microsoft Office
Interactive PROC Report	SAS 9 Programming Enhancements
SQL Basics	SAS with Style: Creating Your Own ODS Style Template for RTF Output
Web Publishing of Reports & Graphs	Format Festival: An Introduction to SAS FORMATS and INFORMATS
Advanced Macros	Traffic Lighting Your Multi-Sheet Microsoft Excel Workbooks the Easy Way with SAS
How SAS Thinks	SAS/GRAPH Elements You Should Know - Even If You Don't Use SAS/GRAPH
Fundamentals of the SAS Hash Object	Introduction to Data Simulation

The number of students depends on the size of the room. Regional conferences, such as SESUG and MWSUG, might have space for 15 to 20 students. SAS Global Forum will have space for many more than that in one room and might have 2 rooms going at once. For popular HOWs, there are two students to each computer. They can take turns doing the exercises or not. If the conference has enough volunteers, there will be assistants to help people with anything usually not directly related to the topic, such as problems with the computer or with SAS, people who get lost or don't know the password. That way the instructor doesn't have to take time away from the rest of the students.

Like all presentations, there is a person, or persons, in charge of the HOWs. They make sure that you get your paper in, the code gets installed and runs, and the room is set up and ready to go. There will also be someone keeping track of time so that you don't go long.

Finally, there will be one, maybe two laptops for the instructor to use for their presentations. Sometimes both the slide deck and SAS are on the same laptop, which may be the instructor's own laptop, or they may be on two laptops with a switch to alternate the projection between them.

In the beginning, a company that did a lot of training sponsored the Hands on Workshops, providing, I think, the desktop computers and the SAS license (in agreement with SAS). Then SAS started to sponsor them, but that became very expensive. Now the conferences are exploring other ways of providing the equipment. Some have tried Amazon Web Services servers to which people would connect from their own computer. Some have bought laptops to use that are lower end, but good enough for the purpose. It's a big expense and it's very good when someone sponsors the HOW. Of course, SAS still provides the software. They are very good about that.

CAN YOU DO A HOW?

Now that you know what a HOW is, or maybe you've been a student at one before, you might want to know whether you could teach one yourself. Good question. To answer it, here are the other questions to consider.

Have you ever taught before? If you have taught before, then that's obviously a big help. I taught for many years at universities and other situations before I did my first HOW. It *might* be possible to do a HOW as your first teaching assignment. The students I have encountered have always been very gracious and forgiving at any mistakes I've made. However, the conference committee owes it to the attendees to provide them a top-notch learning experience. If they don't know whether you can do that or not, then they might not accept your submission.

Do you have time to create the materials? Creating a HOW is much more work than creating a presentation. Besides the paper and presentation, there are the training exercises and solutions. Sometimes finding the right data to use can take time, too. It has to be public, appropriate and a good fit for the topics.

How well do you know the topic? I think that you don't have to know everything about the topic you are teaching. However, you must know a lot. You have a lot of time available, which requires a lot of material. Also, people are going to ask questions that you might never have thought of and it's best if you can answer them right there. Otherwise, you can parking lot it and figure it out later. If you do that too often though, it doesn't look good. Sometimes using the SAS Help to answer the question is good, because it shows people where to find answers on their own.

WHAT TO TEACH

Selecting a Topic

The kinds of topics taught in HOWs is very diverse. Not everything is possible, but a lot of things are. What your topic is should be the convergence of two interests. What do potential students want and what are you interested in. If people aren't interested, then they won't come and you won't be asked back,

assuming you get accepted the first time. If you're not interested in the topic, then why would you teach it? So, the first consideration is the list of topics you find fascinating and/or you know a lot about. Then ask people at work and at conferences, what they would be interested in.

But it's not just the broad topic that you want to know about. Someone may be interested in SAS Macro, but exactly what parts of SAS Macro are they interested in? Do they want an introductory or a more advanced HOW? Do they want to know about creating applications or quoting strings? Because it's almost one and a half hours, you'll have a series of things you'll be teaching them, not just one focused topic.

You also want to consider what the conference organizers would like to see. They are thinking of the attendees too and trying to select the best topics that will draw the most people. There is something to be said for the "standard" topics, such as SQL or Formats. New users may really want these. At the same time, there are already good courses on these topics by experienced teachers. You could give a new flavor or twist to the topic, but it would be good to consider newer topics, too. For example, you could have an HOW on how to use parts of SAS EG or DS2. People are interested in the new things and how they can help them be better programmers or statisticians.

You'll want to make sure that the topic you have and the exercises you do fit with the components and the version of SAS that will be installed on the computers. You can't teach about JMP if it's not installed. Or maybe they restructure the components like they did with ODS Graphics between versions 9.2 and 9.4. Usually the newest version of SAS is available, but it may not be all components.

Exercises

We'll talk more about this below, but realize that because it is *Hands On*, you'll have a lot of exercises. Lecture will be used to introduce complex topics, but you want the student doing much more than listening. How many exercises depends on the complexity of the topic and the exercise. I find that breaking complex exercises into simpler steps makes it easy for everyone to keep up, so I tend to have a lot of smaller exercises. You can also give more or less time for a particular exercise depending on how well people are catching on. However, if you have a lot of exercises and go too fast, then no one will be happy. You might, though, consider having a few more exercises than you will use so that people who catch on quickly can do the "extra credit" and not be bored.

At the same time, you want to minimize the amount of typing that the student has to do. I know it sounds contradictory, but if they have to type too much, then it's easier to make mistakes and the slower typists can get behind quickly. You'll see below that I structure my exercises so that they can either type or copy and paste. And what they do type is a few keywords.

If you've taught before, then you know that the topics/exercises should follow a natural progression. Ideally they will build on each other, so that people can work up to the more complex ideas you want them to know. If you are shifting topics, it's important to be obvious about it. One good way to do that is with a new SAS file for each major shift.

WHAT DATA TO USE

Sometimes what data to use is an easy question. SAS comes with a lot of data sets (over 200) in the SASHELP library. The good thing about these is that they will automatically be there for you.

It may be that none of those data sets have the features you want for your exercises. Maybe they aren't large enough or wide enough. Or maybe you want to show some statistical results and they don't have the relationships you need. Obviously, you can't use company data sets, since they are proprietary. However, maybe you would be allowed to use them if you cleaned them appropriately. Always ask permission of the appropriate management about whether you can use the data or not.

At the same time, there are a lot of public domain data sets that you can use. You might have to search a lot if you are looking for particular features, but remember it's likely not the data that you want to show, but what SAS can do with it.

If you do not use the data sets that come with SAS, then create SAS data sets and distribute those to the conference with your code. (more below) Unless part of the exercise is reading in text files or

downloading files from the internet (an internet connection is unlikely anyway), then it's best to have them ready to go with an appropriate LIBNAME defined in your AUTOEXEC file. The bigger message here is to remove any extraneous tasks, i.e. those not specifically related to the topic of interest.

It's also a good idea to keep the number of data sets used to a minimum, particularly if understanding the data is necessary for completing the exercises. The fewer shifts you have for people the better they will learn.

STRUCTURE AND PROCESS

Let's say you have decided to submit a Hands on Workshop to a conference. It's a topic that you are interested in and know a lot about. It will use software that is available on the HOW computers. Since the Call for Papers (CFP) closes about 4 months before the conference, I highly recommend having a very good start on the paper and the exercises before you submit your abstract. If you can have it done before you submit, so much the better. Then the conference organizers can see your material, which will help them be comfortable accepting a newbie.

The deadlines for the HOW are the same as for other presentations. You still need a Permission to Publish, your final Paper PDF and your presentation. You will also submit your code and other files for them to test and install on the computers. They usually like this about a month in advance, especially if you are new, but that can vary. They will tell you. That really means you only have 3 months after CFP closes to have everything done and ready to go.

Once you create your paper, presentation and exercises, you'll want to try them out in a practice session so that you can work the kinks out. Does everything work? Are the exercises clear to others? Are the solutions correct? How long will the exercises take? Do you have enough or too many? Because of the moving parts, you'll want to prepare more than a regular presentation.

You'll also want to practice moving back and forth between your slide deck and SAS. Not the actual moving back and forth, though sometimes that can get in the way, too, but the timing. How much should you have in the lecture and how much to have in the exercise? Do you do all of the lecture up front and then all of the exercises or go back and forth? What cues will you use to remind you to go from one to the other? I've seen a 50/50 or 60/40 split between exercise and lecture recommended. I recommend at least that, but it depends on the complexity of your topic. Use the slide deck for introductory information, introducing new ideas and for teaching the complex ideas. Use the exercises to teach whenever possible.

Files and Code

You will set up your files and code with a particular structure that is consistent for all of the HOWs. That helps the organizers to test and install, and your students to follow along. It would be good to double check each time what the structure will be, but it is usually something like the following.

C:\how\<your lastname> will be the root folder for your HOW. For me, it would be C:\how\kincaid. See below for files to have in the root folder.

Underneath that root folder, you would likely have

\data   for your data sets. These could be SAS data sets, text files, CSV files, etc., but I recommend having SAS data sets.

\exercises   for the SAS files/programs that your students will open. These will be one file for each main topic. Sometimes people will do one file for each exercise or set of exercises. If you are teaching a more point-and-click HOW (such as for EG), then you might not have any SAS programs files. However, it might be beneficial to have documents that give the steps for them to follow along with screen shots. That way they don't have to wait on you or have something to refer to if they get behind.

\solutions   for the working versions of the exercises. These will help the students if they get lost and you if you forget what you want done.

\results   for any output that you create. You might not have any and that's fine. But you might also have HTML, RTF, PDF, CSV, etc.

\startup   for a shortcut to SAS with references for startup folder, AUTOEXEC file and any custom configuration files. The AUTOEXEC is where you'll put all of the necessary overhead required to prepare the SAS session for the exercises. The shortcut should have your name (and maybe session number) and will be placed on the desktop so that students can start up SAS from the desktop without going into the folders.

Because I've taught graphics classes, I've also used

\images   for the PNG and JPG files that are inputs to the exercise.

You should also have a README.txt file that explains

1. What the Hands On Workshop is about
2. What software (and other) requirements it has
3. Each of the folders and your particular use of them
4. Your name and contact information

If you want to have other folders for other purposes, then you certainly can. However, keeping them to just these few will help the organizers and your students.

It has been recommended to have a one page handout for the attendees to reference. I have never done this. I don't always know how many students will be there and I hate wasting paper. You could have a cheat sheet in your root directory (C:\how\<lastname>) that people could open up on the computer.

It is good to also have a .LOG file in the root folder with the results from running all of your solutions. This will give the organizers or tech support an understanding of what to expect when they verify that all of your code runs correctly. If there is an error or warning that you have either purposefully included or willingly allowed as part of the lesson, then tell them that when you send the materials. That will save some back and forth when it's tested.

Finally, you can, though you don't have to, include a copy of your slide deck in the root folder. That way you have everything in one place. If you are like me and tweaking things until the last minute, then obviously you'll want to bring your own version to the conference.

Once everything is done and ready to go, you will zip it up and send it to the HOW chair. They will have to tell you how they want this done. SAS has their way of getting ZIP files through the firewall. Regional and local user groups will have their own.

HOW TO TEACH

At the Conference Prep

Now that you have prepared everything, submitted it to the organizers and it has been approved, you are on your way to the conference! Make sure that you get there early on Sunday so that you check out the HOW room and the materials. While the organizers are experienced and very good, people do make mistakes and that includes you. There have been times when I've had to reinstall the files on every computer, because of something that I or a tech person did. The good thing is that the organizers and other instructors are very helpful in getting things fixed.

When you check to make sure that everything runs, also remember to delete any output that gets created. You don't want any unexpected names or conflicts for the students. Sometimes that output has come from me when I submitted the files because I forgot to "reset" everything, so check that, too.

Also check the presenter's laptop and configuration. Sometimes you can and want to use your own laptop, but other times it's best to use what's there, especially at SAS Global Forum. Do all of the files run? Is the presentation there and is it the current version? How do you switch back and forth between PowerPoint and SAS? Can all of the students see the projected information well, especially at the bottom of the screen? Walk through your workshop and make sure that everything is as you expected.

The Day Of

Unless you are the first HOW in the morning or after lunch, you will only have a few minutes after you can get into the room before yours starts. As you get ready, find out who the volunteers are that are helping with questions. They are very important and you'll want to call on them if at all possible. You don't want to get stuck helping someone with an off-topic SAS or Windows question while everyone else just sits and waits. If someone has a relevant question, you can answer it. Otherwise, go to the volunteers.

Before you begin (or as soon as you remember), increase the size of the font in your SAS Enhanced Editor and Log windows. That will help people in the back to see better. That also means that you get less on the screen at one time, so be aware of what they can see and what they can't.

As you begin, obviously introduce yourself and the topic. Give the students an overview of what you'll be covering and maybe not covering, so that they have the right expectations. Instead of going right into the lecture, though, get them started with SAS first. That does two things. First, if someone has a problem with their computer or with their SAS installation, then it gives more time to fix it. Second, it gets them "hands on" right away. That's what they are here for. Even if you have to lecture for a few minutes before the real exercises, it gets them into the right frame of mind.

In getting started, I show them how to start SAS, have them check the log file to make sure that everything started correctly and then walk them through the folders, particularly the Exercises and Solutions folders. They will start with the Exercises, but will want the Solutions if they get lost or tired.

Exercises

Once they are oriented to what we are doing, then I give them the introduction into the topic and explain the first set of exercises. Remember that the exercises build on each other, so there might be 3 to 5 exercises in a row that build up to key point. I also explain how they are to do the exercises. An example of two successive exercises that I use is below. I have bolded the comments for this paper. As you know, in the SAS Editor those are colored differently.

```
/*
    Exercise 1.3 ②

①   Add URL's to the points   ③
    Then go to the html file and see the difference in the output
*/

proc sgplot data=who;
*   scatter x=villain y=episodes / <keyword>=url;
    scatter x=villain y=episodes / <keyword>=url; ④  ⑤
run;                                              ⑥

/*
    Exercise 1.4

    Make the points bigger and filled in
    SIZE   FILLEDOUTLINEDMARKERS   CIRCLEFILLED
    MARKERATTRS ⑦
*/

proc sgplot data=who;
*   scatter x=villain y=episodes / url=url  <keyword> ⑧
            <keyword>=(symbol=<keyword> <keyword>=15);
    scatter x=villain y=episodes / url=url  <keyword>
            <keyword>=(symbol=<keyword> <keyword>=15); ④
run;
```

Here are the things to notice.

1) There is a comment that introduces each exercise.
2) Each exercise is numbered. In my scheme, the first number is the set of exercises (each set in a different file) and the second number is the exercise number in that set. These are the 3rd and 4th exercise in the first set.
3) The comment explains what they are supposed to do in the exercise.
4) Exercise 1.4 uses all of the code from Exercise 1.3 and adds other keywords for them to learn. It's not always possible to build on the previous exercise this nicely, but if you can, it helps them, and you, a lot.
5) Most of the code is filled in and they just have to type in certain keywords.
6) The keywords are identified clearly with "<keyword>."
7) The keyword that they have to put in for the placeholder is in the comment in all caps. It's (usually) easy for them to figure out what to put where. On the other hand, it's okay for them to think about it a little. That way, they can also type the code in or copy and paste.
8) Each line of code they have to modify has a "template" right above it. The template is the same line commented out. That way, as they are working through the exercises, they always know what they started with. If they accidently delete too many words or put things in the wrong place, they can use the template to reset.

Sometimes I will put in more keywords than necessary so that they can try out variations. For example, these could be other options to use instead of or along with the ones of focus.

Because my exercises are broken down to simpler steps, I only give them a few minutes for each. If your exercises have to be more complicated, then you can give them longer. Your practice, or someone else going through them, will tell you how long to give them. Either, let them know how long they have, so they know how quickly to work. After a short time, I will work through the exercise on the big screen. This allows them to confirm that they got the right answer or helps them get it if they are stuck.

As they are working on it, I check in with them to see if they are getting it or having trouble. This is another great place for the volunteers to really help. If they see someone who needs on-topic help, then they can let you know so you can answer the question. When it's almost time to move on, check in with them again. If all is good, then go ahead with the next exercise. After some of the exercises you'll want to summarize what they've learned, maybe put it together with some of the other concepts and ask for questions.

As mentioned above, I like to have extra exercises that I don't plan to cover in the workshop. I make sure to tell them about these in the beginning, so that if someone goes through the exercises quickly, they can do these extra ones and not be bored. This "extra credit" could be more advanced concepts or applying the earlier concepts to new situations.

Questions

We have talked about questions a few times above, so we will add only a little more. Sometimes people want to ask questions that are specific to problems they are having in their work. Sometimes you will know and other times it's so esoteric that you will have to look it up or think about it. If you don't know, just say so, and ask them to see you afterwards. If you know and the question is relevant for the entire group, then answer it. If the question is very specific to what they are doing and it's short (a one-liner), then you might go ahead and answer. If it's specific and would take a long time, then you could ask the group if others want to know the answer. If not, then ask the person to see you after the workshop.

Wrapping Up

When the workshop is done, I offer to the students that they can have a copy of the files to use later. Whether you do this or not is up to you. Some people consider this information proprietary. I think it's better for them if they can review the workshop again later. However, I make them contact me and ask me to send it to them. You can either show your information on the last slide or put your business cards out on the table for them to take. Sometimes sending them the files can be tough. Many mail programs won't accept zip files. Sometimes you can change the extension to something else and it will get through,

but Gmail and Outlook usually won't accept even that. Another option is to use a cloud storage location, such as Google Drive or Dropbox, and share the files from there.

Afterwards, you might have people come up with questions. You can answer a few in the room, but usually everyone has to leave the room in between workshops. Other than that you are done! You have successfully completed your first workshop. Nice job!

If there are any changes that you want to make for next time, and the first few times there certainly will be, then mark those. I'll even have a notepad with me so that I can write down those changes or even interesting questions that people ask.

CONCLUSION

Presenting a Hands on Workshop is a very rewarding experience. I really enjoy teaching others techniques and tools that can make their jobs better and easier and in a format that gives them more bang for their buck. These workshops are an essential offering at Global and Regional conferences, too, having been part of them for at least 20 years. Sometimes, though, it can feel like there is a set group of people, almost professionals, who do these and they are the only ones who can. That is partly true, but that's only because few other people create and submit good, exciting workshops. I was able to break into business, so to speak, and while it was a lot of work, it has been very rewarding.

REFERENCES

- http://www.sascommunity.org/wiki/Guidelines_for_Creating_a_Hands-on_Workshop

ACKNOWLEDGMENTS

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Chuck Kincaid
Experis Business Analytics
269-553-5140
chuck.kincaid@experis.com
Experis.us/analytics