A PROPOSAL FOR EVALUATING SAS SKILLS OF POTENTIAL EMPLOYEES

C. Royce Claytor, Virginia Electric & Power Company
Ann S. Allen, Richmond Public Schools
Al M. Best, Medical College of Virginia

Rationale

SAS has become a tool widely used in industry, government, and service organizations as well as research and education. It is in the former sectors that SAS has enjoyed the most growth as "converts" trained in American colleges and universities move out to bring the ease, brevity and enormous capability of SAS to real-world problems. There are many large "DP shops" where the virtues of SAS are widely appreciated and others where it is the primary programming language. Still, there are other shops where someone was hired claiming to be a SAS programmer but what skills he did have made SAS perform far from optimally. The need for desirable, if not required, in the job description.

In this proposal we assume that an "intermediate" knowledge of SAS is strongly desirable, if not required, in the job description for the position in question. Other areas of competence such as managerial skills, proficiencies in other languages, systems analysis, and operations research are not addressed. Other sources of information exist for assessing competence in these areas, and this proposed technique would usually be but a part of an interview addressing all the points encompassed in the job description. Further we assume that the job interview(s) have a limited amount of time to devote to this task and would therefore like to get it over with as soon as possible without running too great a risk of overlooking a promising individual.

From this latter assumption proceeds the assertion that there must be key concepts or experiences which would separate the sheep from the goats better than a random look through the index of the SAS Basics manual.

Goals

Typically, a job interview is not necessarily structured to elicit the optimal information from the interviewee. Too often it is virtually unplanned and ad hoc. The approach we propose provides more consistency in this procedure. Therefore, our goals are to provide:

1. A verbally-administered set of questions or "probes" which will indicate rather quickly whether the interviewee is probably a "novice", "intermediate", or "advanced" SAS programmer.
2. Additional questions within each of these three proficiency levels to refine further the interviewer's judgment of the prospective employee.
3. An "answer key" which does not require an understanding of SAS so that this evaluation technique could conceivably be used by someone with minimal or no knowledge of the language.

To achieve these, perhaps lofty, aims we need an easily administrable set of questions or probes that the interviewer could conceivably ask in a conversation of less than 30 minutes. The questions alone would be no substitute for a face-to-face in-depth interview. Also, it is recommended that at some point at least one competent SAS programmer, someone "advanced" in our description here, should look at examples of the individual's actual SAS code.

We have divided SAS programmers into 3 groups. Below our first group comes what could be called "beginners" or "dabblers". Except for those cases where, a) your salary scale can't even attract a novice, b) you are hiring on the basis of promise rather than experience, or c) you have to hire someone related to the president of the company, we assure that you are not particularly interested in someone of "beginner" grade. The best characterization of a beginner is someone who has read the first few chapters of the SAS Introductory Guide and has actually coded some simple jobs and gotten them to run.

A novice SAS programmer can be described as someone who can get relatively straight forward things done but without much understanding. Such a person will know how to input data from cards, do some simple recoding or creation of variables, sort a dataset, print it, and execute some procedures. At this point it would be very important to note that there are many SAS programmers in university environments who know a great deal about even the most esoteric SAS PROC's but who have little data management experience. In industry the reverse is required in that there is less interest in statistical procedures. Because we believe that the largest need for evaluating prospective employees is in this sector, our proposal addresses this emphasis on data management. In essence, what makes SAS a programming language, what makes SAS different and "good" is its data management capabilities. It is in the use of these capabilities that an "intermediate" SAS programmer is made.

An intermediate SAS programmer can be described as someone who has exercised many of SAS's data management facilities and thinks he/she understands how SAS works. Here we look for familiarity with terminology, technique, and efficiency. There are tremendous gradations here. Our questionnaire should make those gradations clearer. As individuals at this level gain experience and hone their skills, they approach the level of the advanced programmer.
An advanced SAS programmer can be described as someone with wide depth and breadth in SAS, someone who has done a great many things and truly understands how SAS works (as much as anyone does). Of course, one can consider a category even above this (an "oh, Wow!" grade) where the individual knows much more than you do and is correcting your misunderstandings and suggesting cute and efficient ways to do things. Rejecting the likelihood that such an individual would come your way for the moment, an advanced programmer will therefore be identified by knowledge of options and more advanced statements and by style and efficiency.

Description of the Method

To differentiate between these levels of SAS programmers, we have developed a questionnaire to be administered verbally. The instrument is in two parts, the first of which is designed to determine rather quickly which of three grades is probably the most appropriate classification for the applicant. The second part contains three sections, one for each level. If the responses to the questions in Part I indicate that the interviewee is most likely a novice, the interviewer will begin with the questions in the Novice Section of Part II and continue with the more complicated questions until the level of competence is clear.

If the responses to Part I make it look as though the individual is an intermediate, then the questions in the Novice Section can be skipped. The questions in the Intermediate Section of Part II would be more appropriate for determining the individual's level of competence. However, if the easier "intermediate" questions are "failed", the interviewer can drop back to the Novice Section. Likewise, if the interviewee senses that the first assessment was too low, some of the intermediate questions may be skipped (the easier ones) and questions from the advanced section may be used.

Questions in the Advanced Section should be administered if responses to Part I indicate that the individual is above intermediate. Of course, the interviewer can drop back from the Advanced to the Intermediate Section if the first judgment appears incorrect.

Finally, before a discussion of the questionnaire and procedures for administering and scoring it, it is important to remember that even a non-SAS programmer may profitably administer this instrument. In such a case, there would be little or no follow-up and the evaluation might be less precise, but it could be useful in the initial screening of a large number of individuals. We welcome your comments and criticisms as to how well our goals have been met. We also invite you to use our procedure and ask only that you let us know how well it worked or how you modified it to better suit your needs.

Description

The questionnaire in the appendix is divided into two parts. The "Probes" questions in Part I are general questions about SAS and SAS use, designed to discriminate quickly between novice, intermediate, and advanced SAS users. For each question, we provide sample answers which, in our experience, individuals from each grade would give in response.

In Part II, the questions within each section (Novice, Intermediate, Advanced) are arranged in order of difficulty, with the easier ones appearing first, followed by those of moderate difficulty, then the hard ones. Answers are provided for all of the questions.

Procedures for Administering and Scoring the Questionnaire

The items in Part I, the "Probes" section of the questionnaire, may be administered orally in a telephone conversation or in a face-to-face interview. The order of these is of no particular importance; however, all of the items should be administered. The interviewer must have a copy of the answers as well as a means for noting the answer key is especially important if the interviewer has a limited knowledge of SAS.

There are two ways in which Part I may be scored: the tally method or the scale method. Both methods yield a numeric "score" for Part I, which may be interpreted as the initial estimate of an individual's SAS proficiency.

The easiest method consists of counting the number of times an individual makes a "novice", "intermediate", or "advanced" response to the probe questions. This process results in a total score for each level. The level with the highest score is only a superficial indication of SAS proficiency; next, the interviewer will proceed to the beginning of that level's section in Part II for suitable questions. In cases where two levels have the same or similar high scores, the interviewer may wish to begin with the hard questions in the Part II section corresponding to the lower of the two levels and then progress to the easy questions of the higher level.

In the scale method, the interviewer scores each question on a scale of one to three points, with a novice response worth one point, an intermediate, two points, and an advanced, three. A total score for Part I is obtained by summing the scores for each item. Referring to Table 1, the interviewer would be able to determine the initial estimate of SAS proficiency for the individual, and then proceed to the appropriate section in Part II.

If an individual's score is on the borderline between two grades, the interviewer may begin Part II with the hard questions of the lower section and move immediately to the easy questions of the next one.
SAS Proficiency Level Conversion Table

<table>
<thead>
<tr>
<th>Raw Score Range</th>
<th>Estimate of SAS Proficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>17-21</td>
<td>Advanced</td>
</tr>
<tr>
<td>12-16</td>
<td>Intermediate</td>
</tr>
<tr>
<td>7-11</td>
<td>Novice</td>
</tr>
</tbody>
</table>

In Part II, the hierarchical arrangement of sections and items within each section allows the interviewer to begin at any point in the questionnaire, and continue forward until questions are missed, or drop back to easier questions within the same section or in a previous section if the first few are missed. Unlike the questions in Part I, the questions in Part II are grouped within each level. For each degree of SAS proficiency, there are three subsections of questions: easy, moderate, and hard. The questions within these subsections are of comparable difficulty. Not all of the questions for a level have to be asked but all of the questions in a subsection or a level should be asked. The interviewer is free, however, to ask follow-up questions as needed; and, depending on the responses, may skip portions of the questionnaire if it appears as though the interviewee is proficient beyond the level of the questions. Of course, the interviewer must remember that the purpose of the Part I questions was to pinpoint the start of questioning in Part II.

The scoring of Part II involves noting the number of items asked and the number answered correctly in each section, then calculating the percentage of correct answers in each. Table 2 provides a sample of a scoring form which might be used.

The highest grade (i.e. Novice, Intermediate, Advanced) with a percentage correct approaching 100 is the SAS proficiency level of the interviewee. In our example in Table 2, this level is the Intermediate. Even if large portions of the questionnaire are not administered, this method of determining the level of proficiency in SAS is valid: the questionnaire items are arranged in order of difficulty, therefore successful performance at higher levels implies successful performance at lower levels.

A closer review of the data collected in Table 2 further refines the initial assessment of our interviewee's proficiency in SAS. The pattern of response to easy, moderate, and hard items within each section reveals that our interviewee correctly answered all but one of the "hard" items in the Intermediate Section and only one of the "easy" items in the Advanced Section. This indicates that our individual has used SAS's data management capabilities, has mastered techniques for using SAS more efficiently, and is approaching the level of an advanced SAS programmer.

In conclusion, it is not our intent to imply that this single questionnaire in and of itself should be the sole device for providing information about an individual's proficiency in SAS. Certainly an in-depth interview with SAS experts and review of samples of SAS code are reliable tools which should be used as well.

Sample Scoring Form for Part II Items

<table>
<thead>
<tr>
<th>Total Asked</th>
<th>Total Correct</th>
</tr>
</thead>
<tbody>
<tr>
<td>Novice</td>
<td></td>
</tr>
<tr>
<td>Easy</td>
<td>0</td>
</tr>
<tr>
<td>Moderate</td>
<td>0</td>
</tr>
<tr>
<td>Hard</td>
<td>3</td>
</tr>
<tr>
<td>Score</td>
<td>3/3=100%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Intermediate</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Easy</td>
<td>3</td>
</tr>
<tr>
<td>Moderate</td>
<td>7</td>
</tr>
<tr>
<td>Hard</td>
<td>4</td>
</tr>
<tr>
<td>Score</td>
<td>12/14=86%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Advanced</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Easy</td>
<td>4</td>
</tr>
<tr>
<td>Moderate</td>
<td>0</td>
</tr>
<tr>
<td>Hard</td>
<td>0</td>
</tr>
<tr>
<td>Score</td>
<td>1/4=25%</td>
</tr>
</tbody>
</table>

Acknowledgments

The following people were very helpful in the formulation of this proposal: Thomas Blaschke, Sally Carson, Paul Fingerman, Don Henderson, Sandy Hendren, and Mason Nichols.
SAS Proficiency Interview

PART I

Probes

Questions which discriminate quickly between novice, intermediate, and advanced.

1. How many years of experience do you have in using SAS at least once a week?
   a) two years or less
   b) more than two years
   c) over five years

2. Do you read SAS Communications? What do you recall from the most recent issue?
   a) yes; something in the applications and/or technical section
   b) yes; something in the applications, technical, news (new directions), and "people" sections.

3. Have you read the SAS Applications Guide? (purple cover)
   a) yes; something from reading or reshaping data, recoding
   b) of course; something from the above plus one or more of the following: merging, updating, or maintaining SAS datasets; writing reports; working with OS or system datasets; large datasets; SAS code that writes SAS code.

4. In most of your SAS jobs, where is your input data?
   a) following a CARDs statement in my job
   b) in my SAS database

5. What statements allow you to change variable values in a data step?
   a) IF... THEN... ELSE
   b) all of the above plus PROC FORMAT or ARRSYS, DO OVER

6. What's the program data vector?
   a) all the variables in the data step
   b) above plus: with an indicator of which are to be output and what their format is
   c) for each output dataset it indicates: variable name, type, informat, format, label, length and position. (plus some special variables such as _n_)

7. Why do you like using SAS?
   a) strong data management capabilities; self-documenting; multistep design; use of permanent databases

PART II

Novice Section

EASY

n1. What happens if no VAR statement is used in PROC PRINT?
   - All the variables are listed in the order they first appeared.

n2. What character ends SAS statements?
   - ";" (if they say that % ends the MACRO statement they're likely to be at least intermediate).

n3. There are only two types of variables in a SAS dataset. What are they?
   - Numeric and character.

MODERATE

n4. What does the N option of PROC PRINT do?
   - It prints the number of observations (in a by group).

n5. What is the default length of a numeric variable?
   - 8 bytes.

n6. After a CARDS statement, how does SAS "know" when it has reached the end of the data?
   - Encounters a ";" in the next line.
   Typical wrong answer: PROC statement

n7. A BY statement after SORT/MERGE/UPDATE indicates that the input datasets are in what form?
   - Sorted.

n8. What's the difference between INFILE and INPUT?
   - Both are used to read card image type data. INFILE tells SAS where the whole input file is and INPUT tells SAS what variables to read from which columns.

n9. What SAS statements are used to select specific observations?
   - Subsetting IF or IF . . . THEN ... (DELETE or OUTPUT)
n10. How would you write an input statement 2 ways to read variable X from column 2?
- INPUT X 2;
- INPUT @2 X 1;

n11. What SAS statements are used to select specific variables in a data step?
- DROP and KEEP

Intermediate Section

EASY

i1. How is the order of execution of an SAS data step different from the order of execution of FORTRAN (or PLI or COBOL or whatever)?
- SAS executes all the statements in a DATA step for each observation.

i2. What statement(s) are used to concatenate datasets? To interleave observations in datasets?
- SET versus SET with a BY

i3. What symbol is used to concatenate two values?
- Two vertical bars.

MODERATE

i4. If the digits: 010260 were read in with a MONDAYS format, what number is stored internally?
- 1 (2 is acceptable albeit wrong)

i5. How might the result of these two statements differ?
- 1) C = A + B;
- 2) C = SUM (A,B);
- C is missing if 1) either A or B is missing
- 2) both A & B are missing.
- Functions do not propagate missing values.

i6. In a data step, how can you identify which of two input datasets an observation came from?
- IN variables.

i7. How many SAS datasets can there be on a system dataset?
- As many as the file will hold.
- Wrong answer: any specific number or number range.

i8. How would you begin a data step that did not create a SAS data set?
- DATA _NULL_

i9. The BY statement in a DATA step causes the creation of what automatic variables?
- A FIRST. and LAST. for all the BY variables;
- Lets you know when you are processing the first or last observation in a hierarchically sorted dataset (control breaks).

10. What does a two level input dataset name such as OLD,LOW indicate?
- That there is a permanent SAS dataset called 3NO stored on a System dataset indicated by the cdbname OLD.

11. What statement allows you to keep track of previous values of variables or keep a running total?
- RETAIN

12. When an invalid data field is encountered when inputting a numeric variable, what happens?
- 1) A message is printed on the Log
- 2) The raw input line is printed
- 3) The variable is assigned a missing value
- 4) All the variables and their values are printed

13. In the macro language of SAS, all macro commands begin with?
- Right answer - %
- Typical wrong answer - Macro

14. When would you use PROC SUMMARY Instead of MEANS?
- If the data were not sorted by class variables or
- You want statistics for more than one grouping of the data or
- The dataset is big.

15. What's the difference between $w. and $CHARw. input formats?
- No left blank compression in $CHAR (also $CHAR does not translate a . to a blank)

Advanced Section

EASY

a1. How often do you need to compress a SAS data library/file?
- Never; the space is managed by SAS

a2. If you wanted to print a hyphenated phone number with an area code in parenthesis, how would you do it?
- Picture formats.
- (alternative: concatenation of separate variables)

a3. What is the purpose of the double trailing at sign (@)?
- It tells SAS to remain on the same input line for the next observation.
- Wrong answer: to tell SAS that another INPUT statement may be encountered later to reread an input line.
a4. Two datasets have the same variable, X. In a matching observation in dataset A, X equals 3 and in dataset B, X is missing (.). Using A as the leftmost dataset in a MERGE and an UPDATE, would the resulting value of X differ and why?
- In MERGE X = 3, while in UPDATE X = 3 because in MERGE the rightmost value overlays previous values while UPDATE only overlays with non-missing values.

MODERATE

a5. How would you simulate generation datasets?
- PROC DATASETS with the MERGE statement.

a6. What does a SAS dataset contain in addition to the data?
- What procedure will display this information?
  - HEADER information on the dataset (name, when created, tracks & extents, type of dataset, etc.)
  - Attributes of each variable history (source statements)
  - PROC CONTENTS will display the above

a7. What actions are implied by the statement: N=A?
- RETAIN N;
  Initialize N to 0
  N = SUM (N, A); (not N=N+A)

a8. What ddname must be used to access a permanent SAS format library?
- SASLIB

a9. What is N?
- A special variable indicating the number of times the DATA step has been executed (not necessarily the number of observations read).

EASY

a10. How do you control the listing of invalid data in a numeric input field and the error message?
- ?? informat modifier.

a11. How would you build and use a source code or macro library that would allow you to conditionally execute SAS code stored elsewhere?
- Store the modules as members of a PDS and use INCLUDE.

a12. How would you go about selecting certain observations from a rather large SAS dataset if you knew their observation numbers?
- The POINT option of SET allows direct access.

© 1983 Contact Al Best, Box 32 MCV Station, Richmond, VA. 23298. (804) 786-6600 or 786-4533 for permission to use the questionnaire.