

Paper 180-2009

Assessing SAS Skill Level during the Interviewing Process

Jenine Milum, Wachovia a Wells Fargo Company, Charlotte, NC

ABSTRACT

This paper will provide guidelines and tools that will assist in interviewing a candidate for a SAS programming position, even for the non-programming interviewer. Skill level categories will be defined: Beginner, Intermediate, and Advanced. Specific interviewing approaches, suggested topics, questions and tests will be explored. We'll also cover tactics to get the right candidates to apply in the first place.

INTRODUCTION

Interviewing is usually one of those things that many people would prefer not to do; whether you are the interviewer or the interviewee. It requires a lot of mental effort on both sides. There is little or no opportunity to practice without being involved in a true interview. You are either in the situation of needing the perfect new employee or looking for new employment. In the technical world, being the interviewer is usually not one of the skills we had been hired to do in the first place. If you are a technical person, interviewing skills don't always come naturally and requires a certain amount of effort to be successful.

Having a basic predetermined approach to the interview process can yield the required results. Not having an approach may result in misleading information, a vague interpretation of abilities, and be a waste of time for both parties. Having a solid interviewing game plan will provide a better sense of control and confidence. We will consider the approach from a technical perspective and leave other human resource needs and compatibility issues for another discussion.

A certain amount of research is required prior to conducting an interview. Within this research, a set of skills should be outlined. A level of competency needs to be determined for each desired skill. Flushing this information out will provide well thought out guidelines for assessing the skill level you require.

KNOW WHAT KIND OF CANDIDATE IS REQUIRED

Is industry experience a firm requirement for consideration? Is statistical savvy unimportant, but experience with data base management a must? Which operating systems are required? Or, perhaps this is this not a crucial point. What about "soft" skills like business analysis, data modeling, or testing design? Knowing which skills are essential, which are not, and the level of competency each requires is of the utmost importance when conducting a technical interview. Having specific skills identified and expertise levels well defined will allow the interviewer to weed out resumes, bring in only the best candidates for interviews, and devise the right set of questions.

If you're interviewing for a position that is isolated without other SAS programmers or the project is a startup, you may need a candidate with considerable experience or a broader skill set. On the flip side, you may have a real stable programming team or a stable project. In this case, someone with intermediate experience and skills or even a beginner may be ideal. The team can teach and build the skills of a less experienced candidate. Basic understanding of such environments is absolutely crucial.

HOW TO GET THE RIGHT CANDIDATES

What's the best way to avoid wasting time interviewing? The answer is: invite only the best candidates to an interview. It sounds simple, but it is not always easy to do.

Of course, the basic job market climate in a certain region has a big part to play in the search. This matter is for the most part, out of your hands. But there are things you can do, as a hiring manager, to make this situation as unaffacting as possible.

Arguably, the number one way to find the best candidates to interview is by networking. If you have an open position, let everyone you know in the field aware of what you are looking for. No one wants to look bad based on a recommendation. Even someone you don't care much for is likely to only send viable candidates your way. Good developers tend to know other good developers.

One must get the job posting out to as many eyes as possible and to the best markets. Investigate which internet job posting websites are most popular, not only in your region, but for the technically savvy.

Using a job placement firm is a way to save time, albeit an expensive. A good placement company will do a lot of the screening up front saving you considerable time interviewing unworthy candidates whether it is based on financial needs, abilities, or logistics. They will usually place ads in the most effective locations.

Filtering through resumes can be time consuming limit the field. Scanning for key words and phrases before delving into details can quickly bring forward the most appropriate candidates. Yes, occasionally you get a resume for a SAS position and the word SAS is nowhere to be found. In this case there is no sense reading further.

Even using the best of the above suggestions, having a well written job description will be the second biggest way to get the right candidates. Great care needs to go into every word of the description. Be very clear which skills are required. Don't be ambiguous in regard to which skills are preferable but not necessary. Leaving too much subject to interpretation will result in too many inappropriate candidates vying for an interview and you'll end up with a bewildering mess of resumes to review. A description that is too stringent or narrow however might ward off very viable candidates. Minimum experience requirements, specific skills and education level minimums should be very clear.

A well-written job description will deter many candidates that are unqualified from applying in the first place. If possible, have someone else familiar with the position review the job description to determine if the proper description is being conveyed. A solid, accurate description also provides the basis for developing questions and topics to be used during the interview.

TYPES OF INTERVIEWS

There are two types of technical interviews: the face-to-face interview or a phone interview. Both have their advantages and disadvantages.

Meeting a person face-to-face gives a lot more opportunity to assess their skills. The candidate obviously brings to the interview only what they have in their head. The visual aspect can be very enlightening. Confusion, confidence, and comfort with questions are harder to hide in person. Time permitting, programming tests can easily be administered during a face-to-face interview.

There are disadvantages to a face-to-face interview. If you are going to the effort of bringing in a candidate for an interview, it's likely that you will have this person interview with multiple people and possibly incur travel and lodging expenses. Unless the person is a referral, you don't know anything more about the candidate than what is in their resume. You run the risk of wasting the time of multiple people with a candidate unless they can be quickly eliminated from selection (and the interview politely terminated).

Only in rare cases is a technical phone interview anything more than a screening. It allows the interviewer a chance to assess more candidates faster than in the face-to-face interview and doesn't have to involve as many people. This is indeed its biggest advantage. Although, phone interviews tend to take less time they cannot not cover quite as many topics or go into the level of detail needed. Obviously written tests or code review are not really options in this case. This can be partially overcome with a good set of questions that can be answered over the phone.

A series of phone interviews hopefully leads to several good candidates. The next logical action is a face-to-face interview since a more in-depth technical interview is very difficult via telephone. Options such as code reviews and tests should be done at this point. This combination of interview stages, passing a phone technical interview and moving into a face-to-face interview for additional depth, is the most successful.

You can leverage the Internet to learn more about your candidate. Try Google. Enter their full name and possibly "SAS" in the search engine to see what you come up with. If someone says they speak at conferences, there are a number of sites that will show their participation. You just may find a criminal background or some other unsavory information that rules out the applicant. Maybe you find nothing and that may be perfectly fine.

TYPES OF QUESTIONS

There are two types of interview questions: open and closed. A "closed" question is when there is only a yes or no answer or a one word answer. Questions like: "Are you a US citizen?" (yes/no), "How many years have you programmed in SAS?" (#), "What is the SAS numerical value of January 1, 1960?" (Zero).

An "open" question guides the candidate to lengthier responses and explanations. These types of questions

obviously take longer but reveal the most information. It's important to devise questions that elicit honest and revealing responses and require more thought from the candidate. Questions like: "How did you get into SAS programming?" or "Where do you go when you have a SAS programming question?" require full sentence and frequently multiple answers. It's important to remember to give breathing time even after an open question has been answered. The interviewee may often have additional input as they think through their answer.

While open questions reveal the most, they are more tedious on both the interviewer and the interviewee. A nice mixture of both makes for the best interview. The combination of the two types helps keep the flow of the interview moving and the candidate on their toes.

DEFINING BEGINNER, INTERMEDIATE, ADVANCED, AND EXPERT

Definitions of beginner, intermediate, advanced and even expert skill level are as varied as there are jobs. The lines between each are very gray and very much up to interpretation. Someone with advanced skills in one area may be a beginner in another. Or a candidate with few years experience may be a quick learner or have had the opportunity to learn a lot fast. Below is a **general rule** that can be followed with regard to SAS.

Beginner	1-3 years
Intermediate	3-6 years
Advanced	6-10 years
Expert	10+ years

Of course, there are exceptions to everything. With this general time table and presuming you have defined the specific skills you are looking for, you have a basic starting point to begin your assessment. The length of a candidate's experience should be clearly identified in the resume. Based on this you can predetermine a skill level. The responses to the interview questions can be used to confirm their stated experience level. In the best case, their skill level exceeds the general outline above.

SUGGESTIONS DURING AN INTERVIEW

There are some recommendations that can be used to make a technical interview as productive as possible. Some obvious suggestions are "come prepared with a list of questions" and "provide a private uninterrupted time to conduct the interview". Such recommendations are relevant for any interview. Below are some other specific suggestions.

- *Thoroughly familiarize yourself with the applicant's resume before starting the interview. Focus only on topics that are relevant to role being discussed.*
- *Don't hesitate to take notes during the interview. As you go through each question, write down key words used by the candidate as a reminder of their reply.*
- *After finishing a question, it's helpful to categorize the answer with a B (Basic), I (Intermediate), A (Advanced), or E (Expert) for the 4 levels of skill. You can change them later, but it's a reminder of your initial impression. The accumulation of these categorizations will easily highlight your initial overall level assessment of the candidate.*
- *Taking notes and quick assessments during the interview will slow the pace, giving the interviewee time to shift more easily from one question to another. Sometimes the result is additional information from the candidate that would have otherwise been missed by too fast a pace.*
- *If you are an experienced SAS programmer, let the candidate know this up front. This will keep the interviewee from wasting your time giving answers you know obviously aren't correct. If you aren't experienced, do not offer this information.*
- *Avoid leading questions. Stay away from phrases such as "Isn't it right to", "Would you agree that", and "Wouldn't you". All of these phrases suggest an answer. Instead, use phrases such as "List", "Describe", "Tell me about your experience with", "How would you do..."*

- *Use follow-up questions.* When a vague or incomplete answer is given, it might be because the candidate didn't fully understand the question. Coming back with the same question worded differently may yield the desired results. A follow-up question may also be necessary when you think the person is avoiding telling the truth or the full story.
- *Make a specific appointment time for the interview* that is convenient for both parties. You never want to call unannounced to assess a candidate's skill set.
- *Hold your fire.* If you get an incorrect answer, it doesn't benefit either of you to point it out. It will distract the candidate from the rest of the interview. Just make a note of it when assessing the skill level for that particular question.

INTERVIEW QUESTIONS

The following questions are general questions for a general candidate.

GENERAL

1. What version of SAS are you currently using?
 - a. If 9.1.3, then do you know anything about 9.2?
 - b. Don't know, really?
2. What operating systems are you currently running on?
 - a. If UNIX (insert desired Operating System), how long?
 - b. If not UNIX (insert desired Operating System), when, how long ago, for how long?
3. Where did you learn SAS?
4. Do you attend any of local/regional/national SAS User group meetings conferences?
5. Where do you go when you have a SAS programming question?
6. What size of datasets do you have experience working with? What tools do you use to deal with large datasets?
7. How many companies have you worked for as a SAS programmer?
8. If you have a request for a report based on SAS data you have access too, what is your approach to providing the report?
9. How do you QA your results?

SKILL

1. What is the significance of January 1st, 1960?
2. Which SAS Procedures (PROCs) do you use and for what purpose?
3. What is your preferred reporting tool? Why? How does (insert preferred tool) differ from (insert other tool)?
4. List the SAS functions you generally use and how you use them.
5. Describe your familiarity with SAS Formats/Informats.
 - a. Do you create formats? Yes, How?
 - b. What other uses of formats are there?

6. Describe your familiarity with SAS *macros* and the macro language.
7. What SAS system options are you familiar with and for what purpose do you use them?
8. Describe your familiarity with other SAS tools.
 - a. ODS
 - b. EBI
 - c. SAS Administration
 - d. Enterprise Guide
 - e. Other relevant items
9. What have you heard about version 9.2 that you are looking forward to?
10. For what purposes do you use DATA `_NULL_`?
11. Are there other SAS tools you use that you think would be of interest to share?

INTERVIEW QUESTIONS: WHAT TO LOOK FOR WITHIN THE ANSWERS TO THE GENERAL QUESTIONS

1. WHAT VERSION OF SAS ARE YOU CURRENTLY USING?

This is a nice generic question to get the conversation going and is a “closed” question. Only in the rare occasion does the candidate not know this answer. In the case he/she isn’t aware of the version; the candidate would get a big B for beginner.

2. WHAT OPERATING SYSTEMS ARE YOU CURRENTLY RUNNING ON?

This is another generic question to get the candidate comfortable getting started with the interview. What is learned from this fairly “closed” question is often critical but not difficult to answer. The more experienced candidates will likely have used more than one platform.

3. WHERE DID YOU LEARN SAS?

There are quite a few different responses to this question. None of them are wrong. It helps you learn something about the candidate. Was SAS part of a statistics class in college? Did the candidate’s company pay for SAS Institute based training? Is the candidate self taught? What is good to hear is that it was a combination of sources.

4. DO YOU ATTEND ANY LOCAL/REGIONAL/NATIONAL USER GROUP MEETINGS OR CONFERENCES?

There is no wrong answer for this. The candidate may not even be aware there are such things. While this question may be self fulfilling for this author, it’s an opportunity to let them be aware of such opportunities. If the answer is “Yes”, then there is a further opportunity for the interviewer. First of all, if a candidate does attend conferences or local users group meetings, then the candidate tends to be actively working towards increasing his/her SAS skill level. If you as the interviewer know of the groups the candidate attends, there is further potential for a personal referral from such a connection.

5. WHERE DO YOU GO WHEN YOU HAVE A SAS PROGRAMMING QUESTION?

This is a favorite question of mine. Again, there’s really not a wrong answer but if the answer is the “Online Docs” or “SAS Manuals”, I’m disappointed. I would categorize such a person as a beginner. If the answer is “from co-workers and peers”, “SAS-L”, or “on-line conference proceedings” then the person has likely got more experience. But more importantly, the candidate with those answers is likely to learn faster.

6. WHAT SIZE OF DATASETS DO YOU HAVE EXPERIENCE WORKING WITH? WHAT TOOLS DO YOU USE TO DEAL WITH LARGE DATASETS?

Everyone has a different impression of what is considered a large dataset. Even if a candidate hasn’t worked with large datasets, they should be able to come up with some creative answers. The complexity and variations to this question will give you a pretty good feel for the candidate’s skill level with SAS datasets whether it fits your needs in this area.

7. HOW MANY COMPANIES HAVE YOU WORKED FOR AS A SAS PROGRAMMER?

While it is never good to hear a candidate has job hopped significantly, it can be a good thing for an experienced programmer to have worked under multiple situations. You can learn an interesting amount about a candidate based on which companies he/she has worked for.

8. IF YOU HAVE A REQUEST FOR A REPORT BASED ON SAS DATA YOU HAVE ACCESS TOO, WHAT IS YOUR APPROACH TO PROVIDING THE REPORT?

Some may suggest exporting the data to Excel or another non SAS tool to create the report while others will summarize the data and create the reports and/or graphs using SAS tools. A more experienced SAS developer during an interview should suggest using SAS tools.

9. HOW DO YOU QA YOUR RESULTS?

An answer to this should come easily even if the responses are quite varied. Do they have previously published results available which should tie directly back to their results? Is their experience with the topic extensive enough for them to have a general idea of reasonableness? Do they do pre summarizations that will lead to an overall confidence level? Code and result reviews in some situations are required. You will know your working environment and what is expected.

WHAT TO LOOK FOR WITHIN THE ANSWERS TO THE SKILL QUESTIONS

1. WHAT IS THE SIGNIFICANCE OF JANUARY 1ST, 1960?

This is a nice transition question. A candidate past Beginner should be able to answer this question with ease.

2. WHICH SAS PROCEDURES (PROCS) DO YOU USE AND FOR WHAT PURPOSE?

This is a very Open question. An experienced programmer will be able to list a dozen procedures. A beginner or intermediate candidate may only list about five and then start repeating themselves. The complexity and variation of the procs mentioned will give you a good bit of insight. I recommend letting the candidate list all the procs they would like. Then go back over a few of the procedures and ask for what purpose they were used. This will flush out the ones that may be “fluff” answers rather than actual procs they have experience with.

3. WHAT IS YOUR PREFERRED REPORTING TOOL? WHY? HOW DOES (INSERT PREFERRED TOOL) DIFFER FROM (INSERT OTHER TOOL)?

There are lots of answers to this question. A less experienced candidate will usually list PROC PRINT and PROC FREQ. A more experienced programmer will start listing all the reporting tools they've used, PROC TABULATE, PROC REPORT, DATA _NULL_, PROC OLAP, PROC GCHART and on and on. They should be able to explain the advantages and disadvantages of each procedure as well as the appropriate situations in which to use them.

4. LIST THE SAS FUNCTIONS YOU GENERALLY USE AND HOW YOU USE THEM.

This question more than any other seems to trip up all but the very experienced candidates. If the interviewee seems to struggle with what a SAS *function* is, give them an example. I usually suggest SUBSTR to get them going. If they can only come up with a couple more, or start listing things that aren't even functions, the candidate is likely a beginner. The more advanced candidates will list so many you'll want to stop them. The types of functions listed will give you good insight into the types of applications they've worked on in the past and help determine if it's a good fit for your needs.

5. DESCRIBE YOUR FAMILIARITY WITH SAS FORMATS/INFORMATS.

There is a wide range of answers to this question. A beginner will usually list some date formats. Engage the candidate into when they use them and how. Ask the candidate whether he/she creates formats and if so, for what instances and by what methods. If a candidate only lists PROC FORMAT, then they tend to be on the less experienced side. Only an advanced/expert level candidate will be able to list other uses for formats and have used data steps and CNTLIN statements to create them.

6. DESCRIBE YOUR FAMILIARITY WITH SAS MACROS AND THE MACRO LANGUAGE.

This too is a wide open question. A beginner usually will know how to call macros and suggest they are used for procedures that are being repeated. A more advanced candidate should be able to carry on a lengthy conversation about macros, how they're used, when they use them, and how to deal with code within macros. Many candidates will confuse macro language with macro variables. This is another good indicator of experience level.

7. WHAT SAS SYTEM OPTIONS ARE YOU FAMILIAR WITH AND FOR WHAT PURPOSE DO YOU USE THEM?

There are over 300 SAS system options. It's not likely you know all of them. A beginner may not even know what you are talking about, or will only be aware of frequently used ones such as *MISSING=* or *OBS=*. More experienced programmers will be quite familiar with the macro options while advanced users will know and use environment options. The more options an interviewee can clearly explain and the variety of activities they're used for will show the level of depth of their SAS knowledge.

8. DESCRIBE YOUR FAMILIARITY WITH ODS, EBI, SAS ADMINISTRATION, ENTERPRISE GUIDE, AND MORE.

A beginner may never have heard of anything more than Enterprise Guide. If the candidate expresses knowledge about any of these tools, then delve into asking details and how these tools were used and/or applied.

9. WHAT HAVE YOU HEARD ABOUT VERSION 9.2 THAT YOU ARE LOOKING FORWARD TO?

Most beginners are probably not aware of version changes and updates, while an experienced programmer will likely be able to tell you a good deal about the next release. Yet, if a candidate only has minimal years of experience but knows about the enhancements coming in the future, this shows real ambition.

10. FOR WHAT PURPOSES DO YOU USE DATA _NULL_?

Beginners to Intermediate might only suggest using DATA _NULL_ to create macro variables. A little more experience would have a programmer describing output to a text file. They might also suggest outputting to html or using them for testing. There are lots of responses here, combinations of answers rather than just one is truly where the skill level is flushed out.

11. ARE THERE OTHER SAS TOOLS YOU USE THAT YOU THINK WOULD BE OF INTEREST?

Once the expect skill set areas have been covered, this is a final opportunity for the candidate to talk about additional SAS skills not specifically covered. A beginner may have little or nothing additional to add. An advanced candidate should use this opportunity to list other SAS areas for competency to show their depth.

Likely, as the interviewer, you would have additional questions about skills specifically needed for the position available. Cover them as much as necessary and keep to an open format.

AN IN-HOUSE TEST

This is a test that is to be given during a face-to-face interview. It is designed to allow the candidate a chance to "show off" and is applicable to all skill levels. Cleverness in approach while still giving correct results is the most sought after answer. This coding task can reveal a great deal about a person's skill level. Make it clear that you are not looking for the perfect answer, but looking for creativity. 10 minutes should be more than ample time for a candidate to complete this task.

Using SAS, combine Dataset A with Dataset B. Keep only those records that are contained in both A & B. Creativity rather than the perfect answer is being sought.

<u>Dataset A</u>	<u>Dataset B</u>
ordernumber	ordernumber
firstname	product
lastname	purchase date

Having given this test to quite a range of candidates and associates, I have been surprised at the uniqueness of answers each person submits.

At the very least the answer should be:

```

/* Sort-Sort-Merge */

proc sort data=dataseta;
  by ordernumber;
run;

proc sort data=datasetb;
  by ordernumber;
run;

data both; merge dataseta(in=a)
              datasetb(in=b);
  by ordernumber;
  if a and b;
run;

```

If these results aren't at least mastered, then the candidate truly is no more than a beginner at best. Other more competent and clever answers have included macros, SQL, and PROC FORMAT to get the same results. These types of responses will show greater understanding of the SAS *programming language* and the ability to think outside the box.

Here are a few previously submitted answers and an assessment as to their skills:

```
/* Merge */

data both; merge dataseta
              datasetb;
  by ordernumber;
run;
```

Unfortunately, I've gotten this answer before. You can deduce either the candidate knows very little SAS or they did not read the instructions nor listen to your expectations of desired results. In either case, it's not likely that such a candidate would make a good SAS programmer.

```
/* SQL */

Proc SQL;
create table result as
select a.*, b.*
from a, b
where a.ordernumber=b.ordernumber;
quit;
```

Quite a few beginner SAS programmers already have experience with SQL. This answer doesn't necessarily impress me unless evidence of previously declared experience with SQL is needed.

The two previous answers get the job done but lead you to believe they haven't had much experience and probably haven't been exposed to many other people's code.

```
/* macro */

options symbolgen;

%macro doit(ds1,ds2,sortvar,mergetype);

proc sort data=&ds1;
  by &sortvar;
run;
proc sort data=&ds2;
  by &sortvar;
run;

data match; merge &ds1(in=a)
                  &ds2(in=b);
  by &sortvar;
  mergetype = &mergetype;
  select(mergetype);
    when (1) if a and b;
    when (2) if a;
    when (3) if b;
    when (4) if a and not b;
    when (5) if b and not a;
    otherwise;
  end;
run;

%mend doit;
%doit(dataseta,datasetb,state,1);
```

This is a clever answer. It shows the person is already thinking ahead a few steps, not just trying to solve the problem at hand, but trying to create a useful module for their SAS "toolbox". This person has probably worked in a very structured team environment where re-usable code is the standard. This is certainly an advanced programmer.

```

/* Proc Format */

data b; set b;
  start = ordernumber;
  label = '*';
  fmtname = '$key';
run;

proc sort data=b nodupkey; by start;
run;

proc format cntlin=b; run;

data all; set a;
  if put(ordernumber,$key.) = '*';
run;

```

This answer is my favorite. That means they've read my paper and have applied it. There aren't many people that come up with this approach off the top of their heads. They've usually seen it done and tried it. Such results likely means they pay attention to new techniques and try to apply them.

Occasionally an interviewee will list several options, possibly using SQL or a hashing routing. One being the method they would most likely use and then another set of code to "show off". I am impressed when this happens as it shows the candidate wants to make sure he/she has clearly delivered what was asked.

SAMPLE CODE

I like to ask a candidate to bring in a piece of a working program they had written of which they are proud, illustrate their skill level(s), and typify the work they're doing now. Review the code with the candidate. This isn't a detailed code review, but has the candidate explain the general flow and to point out items in the code they think are of value or are important. Ask questions about items the candidate speak about or ask why they chose a particular technique or approach. This is to make sure the candidate really knows the code. Of course, it's possible the code was written by someone else and you must ask tough questions that will rule out (or confirm) any doubts you might have. They must demonstrate a full understanding of what the code is doing and why. Pay particular attention to comments in the code (or lack thereof), and see if the code follows an easily readable style.

Even if you're a non-SAS interviewer doing the interview, the candidate must communicate a strong sense of understanding. If the candidate is interviewing with others on the same visit, don't hesitate to come back later with additional questions about the code. Time used during an in-house test is ideal to review the sample code.

WRAPPING UP A TECHNICAL INTERVIEW

After all the questions you've prepared or thought of on-the-fly have been thoroughly explored and you feel you have enough appropriate information to make an assessment, allow the candidate some time to add additional information they feel may be appropriate and of interest to you. Not only is this polite, occasionally there is interesting information gathered.

An experienced interviewer will be able to avoid too much feedback to the candidate during the interview. If the goal of the interview is truly to extract technical skill level, letting the candidate know if they were right or wrong in response to questions should be avoided so that the candidate is not "lead" in a particular direction. Even in a more general "getting to know you" interview it's not a good idea to let the candidate know what you're looking for—you need to see the real person you could be working with, not an adopted persona.

I have found that many interviewees want some kind of feedback and sometimes even ask for it. An interviewer should always be prepared for such an event. A simple "I've enjoyed speaking with you" or "I'll be letting my manager know how our discussion went and they'll be getting in touch with you if they want to proceed further" are ways to avoid giving them too much information or committing. These phrases are also good to use to wrap up a technical interview by letting the candidate know that you are through with questioning and have nothing further to add. It's very important not to inadvertently set inappropriate expectations based on your response to the "when will I hear something?" question.

Make sure to retain a copy of every candidate's resume along with your final assessment, notes, summarization of skills and the job description. You never know when someone might ask if you know anyone looking for a job (remember what I said about networking?) or a new position might open up for which the interviewee may be well suited. It's a good precaution for anything unexpected rather than having to relying on your memory, such as your first choice does not to accept your offer and you need to reopen the search.

CONCLUSION

There are as many different interview questions that could potentially be asked as there are words in the SAS Procedures Guide. Many people have quite different approaches to the process that work for them. There is no one perfect way to perform proper skill set screening. The approach outlined in this paper is what I have found to be highly successful as it has evolved over the years through much experience performing technical interviews.

It takes time to become a competent "SAS programming skills set" interviewer. The more often these questions are asked and the more familiar one becomes with the varied responses, the more confident an interviewer will become with the results. It doesn't require the interviewer to be a SAS programmer. Rather it requires the right "ear" for key words and responses.

A combination of technical questions, both "open" and "closed" will illuminate the required information. It's important to stick to a selected set of questions amongst all candidates to evenly determine the results. It will also keep you, as the interviewer, on target and help you maintain confidence. A well thought out set of interview questions is part of the battle.

The above questions and tests will provide tools and directions to get through any interviewing situation. However, the need to personalize the interview should not be minimized. The absolute most important result to come out of any interview, whether it's face-to-face or a phone interview, is "will this candidate best fit the role that is available". The requirements of any SAS professional position can vary so much, that knowing what you need is the first step. Knowing if a candidate can fit those needs is the final step.

ACKNOWLEDGMENTS

Thanks to John Bentley and Carla Mast, for your considerable editing skills and valuable input.

CONTACT INFORMATION

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

Name: Jenine Milum
Enterprise: Wachovia a Wells Fargo Company
Address: 200 S. College Street
City, State ZIP: Charlotte, NC 28244
Work Phone: 704-383-7377
E-mail: jeninemi@yahoo.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.