

Paper 148-30

Fast and Easy Ways to Annoy the Job Security Specialist

Rick M. Mitchell, Westat, Rockville, MD

ABSTRACT

While SAS programmers and statisticians have been spending their days annoying each other with techniques that are often unethical or just downright nasty, the job security specialist has quietly snaked along under the radar. The job security specialist is one who is fearful that others could potentially steal their precious SAS code and thus put the wheels in motion for either the shifting of work or even replacement of his or her position. Such behavior often leads to decreased productivity, dissatisfaction in the workplace, and prevention of an otherwise very capable and competent team of programmers from meeting their full potential. This paper will discuss fast and easy ways that both SAS programmers and SAS programming managers can reverse the table and have fun annoying the job security specialist.

INTRODUCTION

Definition of a Job Security Specialist – A SAS programmer who hopes to be heavily relied on and ultimately preserve his or her job as long as so desired by intentionally acting deceptively, writing SAS code that is difficult or impossible for others to understand, and/or sharing little or no information regarding his or her SAS programs and corresponding SAS output.

Convincing “special” SAS programmers to open up their world to others can be an extremely difficult and frustrating effort. Programmers and managers have been forced to deal with this problem since the inception of SAS. Without a clear understanding of what makes these “special” programmers click, efforts to beat the job security specialist can be extremely difficult, if not impossible. It is not so uncommon for coworkers to give up quickly on trying to defeat these individuals. For those who are persistent, the end result is often only anger, frustration, and annoyance. This paper will lay out a fast and easy plan that will hone in on and annoy these job security specialists to change both your world and theirs forever.

OVERVIEW

This paper will discuss the following topics:

- How Does the Job Security Specialist Fit Into the Team?
- Why is the SAS Job Security Specialist Special?
- Reeling in the Job Security Specialist
- Creating an Environment Where the Specialist Can't Survive
- Bringing the Job Security Specialist Onboard

THE FAST AND EASY TIPS

“Fast and Easy” Tips are provided throughout this paper not only to spark thought and discussion, but most importantly, to help you as the SAS programming manager or the SAS programming co-worker get your own personal job security specialist onboard with a winning approach. While these tips are not guaranteed (use at your own risk!), the author feels that one may significantly increase one’s chances of not only defeating the job security specialist, but perhaps even helping this individual “see the light” and turn negative actions into positive ones. If you are one who likes to get to the bottom of things quickly, then this paper and its tips could be what you are looking for in your frustrating journey to defeat the job security specialist. Rid your workplace of the job security specialist by implementing these fast and easy tips into your everyday work environment!

HOW DOES THE JOB SECURITY SPECIALIST FIT INTO THE TEAM?

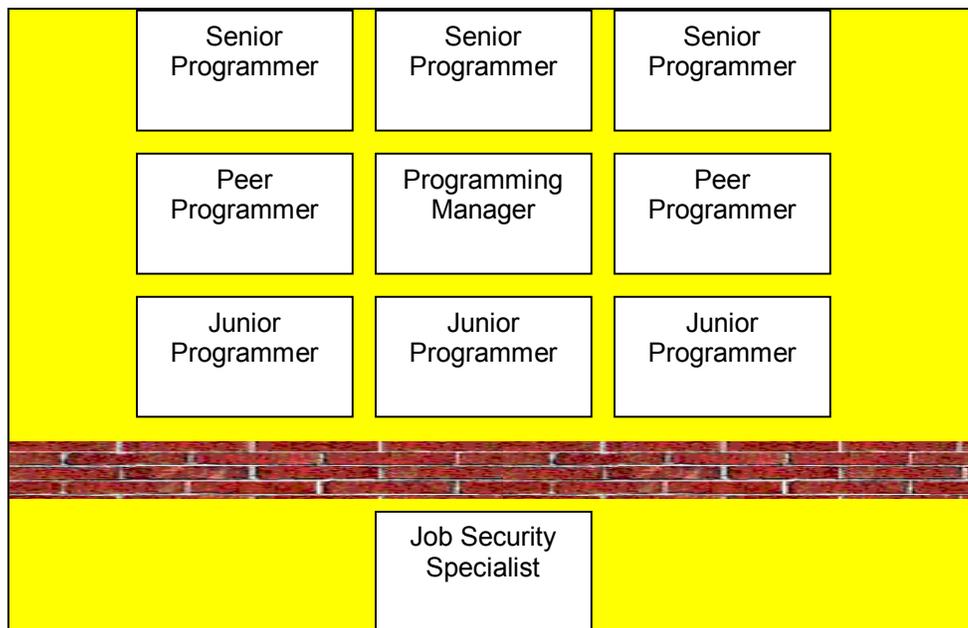
Fast and Easy Tip #1

Give an ultimatum ... Mr. Job Security Specialist, tear down that wall!



The job security specialist can affect a wide audience, and whether one is a SAS programming manager or a SAS co-worker, the job security specialist's role within the team is very much different than his or her peers. This individual can affect nearly everyone on the team either intentionally or unintentionally by merely going off and doing his or her "own thing." While the job security specialist may enjoy working entirely independently, he or she ends up being shut off from the rest of the programming world. Any chance to interweave within the group is lost, and the job security specialist actually sets the environment for his or her own downfall as severe limitations are placed on growth opportunities. Let us examine the structure of a hypothetical SAS programming team as shown below. One can see that the job security specialist is clearly separated from the rest of the team and has no harmonization with the structure and

Hypothetical Structure of a SAS Programming Team



cohesiveness that has been established with the rest of the team. Regardless of the programming level of expertise and the hierarchical structure of the team, each programmer in this team relates to the other programmers in some manner. Note that the SAS programming manager is not necessarily the biggest "hot shot" on the team. Programmers with equivalent skills (peers) may be on hand as well as more senior level programmers who are available to do fancier and/or more complex code (e.g. heavy duty statistical programming). Junior level programmers perform other tasks that help form the foundation of the SAS programming team. The close working relationship between all of these programmers allows greater opportunity for learning and growth for everyone. There are no limitations in terms of which programmers can learn from the others. Senior programmers can sometimes pick up some nifty tricks from junior programmers, and junior programmers can learn from the guidance and examples of senior programmers. All programmers are mixed together to form a solid working team, and regardless of the setup of this team or the number of members, the job security specialist often creates a strong obstacle in the form of a brick wall that is often difficult if not impossible to break through. The longer that the team waits to reel in the job security specialist to their own world, the stronger the brick wall becomes and the lesser chance that it has of ever being broken.

The Ultimate Goal – Break down the barrier and integrate the job security specialist into your SAS team!

WHY IS THE SAS JOB SECURITY SPECIALIST SPECIAL?

Fast and Easy Tip #2

Put up a sign in your parking lot, "Job Security Specialist Parking Only!"



At first glance, one may read the title of this paper and wonder how the SAS job security specialist could be any different from a job security specialist from any other programming language. In many ways this assessment is correct since the main focus of this paper is basic theory rather than code. In fact no SAS code will be presented in this paper, so one could easily assume that he or she could merely present this paper at any programmers' conference and just change the name of the programming language at the top of the paper. However, there are several key characteristics of the SAS programmer that are indeed unique to the programming world and when combined together these can result in a breeding ground that further fuels the job security specialist. These characteristics by themselves are quite positive to nearly all SAS programmers, but when all of them are combined, they have been known to create an environment in which the job security specialist is very, very AFRAID.

AFRAID

Active – There are few programmers in the SAS world that are not active to some extent in the sharing of information. From SAS brown bag lunches within their respective companies, to local and regional SAS conferences, to the big SUGI conference, SAS programmers are most often eager and willing to share and learn from each other.

Friendly – Plain and simple, most SAS programmers are very friendly people. They are willing to share their SAS code, approaches, and toolbox of tricks with their coworkers, their peers, and their fellow conference attendees. However, this can further infuriate the job security specialist who aims to develop code that no one else knows about.

Resourceful – SAS intentionally gives incredible flexibility to its programmers such that even the least resourceful programmer should be able to accomplish what he or she sets out to do. For the advanced SAS programmer, resourcefulness is essential if one wants to take a program to the highest limits and expectations of the SAS world. All of the necessary tools and tricks are out there and just need to be discovered by the resourceful SAS programmer.

Adventurous – As the famous US Army slogan goes, "SAS programming is more than just a job, it's an adventure!" Oh how true this is. SAS programmers don't just go into work each day and type away meaningless code. There is a special enthusiasm and sense of adventure amongst SAS programmers who routinely spark endless communication and excitement throughout the SAS community. It is common for one to "boast" about new tricks that are learned, and programmers frequently share this information so that others may benefit and grow as well. SAS programming is a daily buildup of knowledge in which there is no foreseeable limit. It is this environment that makes the true SAS programmer the true adventurer.

Independent – It actually is okay to be an independent programmer and do one's own thing. SAS is indeed a great environment for programmers to run off and attack various tasks without huge teams of coworkers. This is all fine as long as the independence applies to the task itself, and not to the environment in which the programmer works. Using SAS both independently and as a team leads to a winning formula!

Daring – Because of the extensive range of tools that SAS provides, it is not so uncommon for SAS programmers to want to take SAS to its limits. Few SAS programmers settle for the basic examples provided in those wonderful SAS documentation guides, and they more often try to expand their horizons daring to take on any SAS programming challenge that comes their way. To dare is to achieve!

The job security specialist is always on the move in fear that others will move in on his or her territory. An advanced programmer who has done this for many years is even more dangerous as they continuously come up with clever ideas to ward off others. Some techniques that a job security specialist has been known to keep under his or her belt might include:

- Deleting programs and/or datasets from the network and storing them on their personal drives
- Encrypting all programs and datasets on the network to prevent access by others
- Writing cryptic code that others will have trouble understanding
- Writing GOTO statements that go nowhere
- Including extra code that is meaningless to the program
- Sabotaging coworkers' programs

If you or someone in your company is experiencing the effects of such a person, then it is time to develop a plan and begin reeling in the job security specialist!

REELING IN THE JOB SECURITY SPECIALIST

Fast and Easy Tip #3

Get all of your fellow SAS programmers to join you in confronting the job security specialist with an Intervention!



The 1st step in getting the job security specialist to join your cause is to gain his or her trust. In all likelihood, a constant combative approach is just not going to work. In fact, this may infuriate the specialist even further resulting in an even larger communication gap between the job security specialist and your team. You ultimately want the specialist to join your side, so you need to figure out how to convince him or her that your direction is the right direction. If everything goes smoothly, then this “Reeling In” process can help set the foundation for a potentially long lasting and productive relationship with the job security specialist. In order to make this happen, a non-threatening approach that one may want to try is the **SAFE** approach as described below.

SAFE

Sensitivity – Okay, sure the job security specialist looks like a bad guy (or gal) on the surface, but we all have feelings. Regardless of the approach that you take to reel in this individual, always consider these matters extremely sensitive and consider how your actions will or will not affect the big picture.

Accommodation – If you consider the job security specialist worthy enough to reel in, then you are going to have to accommodate him or her to some extent. Those who take the “black and white” approach and never consider any gray area are just setting themselves up for failure. Determine issues in which you can meet the job security specialist half way and remember to pick your battles!

Facilitation – Unless someone takes the initiative, nothing is going to change. Facilitate an atmosphere that sets standards, yet still gives the job security specialist room to breathe. Remember, the ultimate goal is to get the job security specialist to buy into your program and come onboard, not total alienation!

Expectations – Make your expectations known from the start. Unless the job security specialist understands the importance of what you are doing, he or she is going to go further back into his or her shell. All plans should be clearly laid out and there should be no dramatic surprises down the road.

Don't let the job security specialist get to you. This first phase of “reeling” this individual in can be extremely problematic, so one needs to make the job security specialist feel as **SAFE** and comfortable as possible. Once this **SAFE** environment is established, then you can begin reeling in!

CREATING AN ENVIRONMENT WHERE THE SPECIALIST CAN'T SURVIVE

Fast and Easy Tip #4

To find out exactly what the job security specialist is doing, take away all network access rights and you will soon find out.



A SAS friendly environment allows for endless opportunities, but only if everyone on the team buys into this concept. In order to make this happen, you will need to introduce the job security specialist to this environment and once the brick wall barrier begins to crumble, it will be impossible for the job security specialist to continue to stay under the radar. An environment in which everyone is part of the team and everyone benefits from the SAS team knowledge base is a definite recipe for success. The ingredients to develop and implement this type of environment are certainly not new to the programming world, but perhaps presented in this manner within the SAS programming arena, one can better absorb this information and hopefully take action that will ultimately result in success. It is suggested that this type of environment might include the following mandatory elements:

- Participation in the team repository
- Peer review of SAS code and deliverables
- Naming conventions and storage structure for all electronic project files
- Backup assignments for all tasks
- Documentation procedures for all SAS programs

Participation in the team repository

Fast and Easy Tip #5

Develop a SAS “repository” with everyone’s SAS work so that all may reap the benefits.



Imagine a central location that contains a wonderful arrangement of helpful hints, tricks of the trade, and other valuable resources that could benefit a wide range of programmers. Such a SAS “repository” can not only help drive the growth potential for a company’s SAS programmers, but it can also help prevent programmers from having to waste efforts on reinventing the wheel. Programmers generally use the current tools and knowledge that they know of at the time, and they are often unaware of some of the approaches that can be taken in order to get exactly what they need with SAS. Even the smallest SAS trick may not appear to be an earth shattering discovery, but it can ultimately lead to many hours of saved effort on the part of the SAS programmer. Such a repository has been a success at Westat (contents are shown below) with its entire SAS programming team having a tremendous wealth of knowledge available at their fingertips.

SAS Resources



- [View SAS Online Documentation](#)
- [Order Printed SAS Documentation](#)
- [SAS FAQ](#)
- [Westat SAS Classes](#)
- [Westat SAS Users Group](#)
- [SAS Conference Papers](#)
- [SAS Macro Library](#)
- [SAS Tips](#)
- [SAS Fact Sheets](#)
- [SAS Product Availability](#)
- [SAS Outlook Information Forum](#)
- [SAS Links](#)
- [Version 8 Information](#)

Programmers may not always feel entirely comfortable sharing their work either because they don't feel very confident with what they produced (they may be worried that others will find mistakes) or they are worried that others will steal their precious tricks that make them stand out from others. If your team is able to overcome these issues, then a SAS repository will be advantageous to all and allow the SAS programming team to reach new heights of success.

Peer Review of SAS Code and Deliverables

Fast and Easy Tip #6

Implement the Peer Review Process within your SAS programming team.



While some may view a peer review process as invasive or intimidating, the process should be seen in a positive light as it allows for enhanced feedback, confirmation that a task is on track, and the distribution of knowledge for everyone's benefit. When faced with projects where multiple players are involved, a peer review process can be extremely beneficial. The author worked on such a project where a database was utilized that was forever rich with analysis options and it covered a wide range of disciplines in clinical research. In order to maintain as much standardization and consistency as possible for the 50+ analyses that were going on at the same time, requests came into a data operations center (DOC) via a central mechanism. Any data and/or output coming into or out of the DOC required extensive review, and without the peer review process, this project certainly could have resulted in disaster. No more flying under the radar - the peer review process allows everyone on the team to know exactly what's going on and pretty much forces the job security specialist to reveal any tips or tricks that have been put in place. This peer review process might include:

- A summary of what is being presented
- All corresponding SAS code
- The basics – frequencies, listings, and mean statistics
- Intermediate output (e.g. programming checks)
- The deliverable

While all of the items are important in the peer review process, they could be worthless in terms of defeating the job security specialist unless they are appropriately reviewed. The review of these documents should include:

- Gaining a basic understanding of the requirements and specifications, how and why the code approach was taken, and whether or not the deliverable meets all of the requirements.
- Confirmation that all derived variables have supporting output to document their correctness
- Identifying cryptic code that needs to be explained and/or documented
- Addressing questions such as how would future additions, deletions, and/or enhancements to the program impact the manner in which the program is currently set up

Naming conventions and storage structure for all electronic project files

Fast and Easy Tip #7

Set mandatory rules for what SAS programs are called and where they are stored.



When a programmer is working solo, one can easily slack off on standards and procedures since they really don't appear to affect anyone else, at least on the surface. The programmer may have everything up in his or her head in terms of knowing where things are located, what they are called, what they do, etc. Not only can this be problematic considering that a backup person may need to take considerable effort to catch up to speed if necessary, but if this was the approach within a big programming team, then this could be a nightmare! In order to avoid confusion and to give programmers the opportunity to identify information quickly and accurately, standard naming conventions and storage policies are essential.

Naming

The names of programs and datasets should always give sufficient information to the reader regarding the overall purpose. When naming these, one should always ask the following questions:

- Does the name provide information about the project?
- Does the name provide information about the task?
- Does the name provide information about the date or version number?

If naming conventions are appropriately followed, then they will disarm the job security specialist's ability to cryptically store information.

Naming the SAS Program	
<u>Job Security Specialist</u>	<u>SAS Programming Team Player</u>
ZYXWVUT.SRQ	KITE_BUDGET_BY_DEPARTMENT_2005.SAS

Naming a SAS Dataset	
<u>Job Security Specialist</u>	<u>SAS Programming Team Player</u>
ZYXWVUT.SRQ	KITE_BUDGET_BY_DEPARTMENT_2005.SAS7BAT

Storage

There are 2 rules that should always be followed in the team-programming world. Rule #1 is "ALWAYS store your programs, documentation, and data in a central location that all can access." If you are ever, ever in doubt of whether this rule should be followed, then proceed to Rule #2 which states "Please refer to Rule #1!" In the figure below, one can see that there are 2 key factors in storing the information: (1) the information is stored on a network drive, and (2) sufficient information is provided to describe what is stored there. Such storage methods will disarm the job security specialist's ability to hide information from the team.

Storing the SAS Information	
<u>Job Security Specialist</u>	<u>SAS Programming Team Player</u>
C:\MY DOCUMENTS	P:\FRANKLIN_KITES\BUDGETS\2005\QUARTER1

Backup assignments for all tasks

Fast and Easy Tip #8

Assign a dedicated person as backup support for every SAS task.



Even if there is no given job security specialist lurking around in your company, there should always be an assigned backup person for any SAS programming task. Having a backup will have the following advantages:

- Work will not come to a dead halt if someone is on vacation, out sick, or unavailable because of other pressing projects needs
- Others might benefit from the knowledge that was developed
- Another set of eyes may help address potential issues or even result in improvements
- Limited transition effort when someone leaves the company

The environment in which the peer review process was discussed earlier in this section covered so many different analyses within the same project that backup support and coordination of overlapping or similar issues was essential. The project included:

- 40 published manuscripts over a 5-year period.
- Multiple investigators
- Multiple programmers
- Multiple statisticians
- Multiple analysts
- Multiple protocol team members
- And many, many more!

Without the ability to allow programmers to provide cross-coverage and backup support, it would have been extremely difficult if not impossible to manage such a complex project with constantly changing needs and priorities.

Documentation procedures for all SAS programs

Documentation, documentation, and documentation! Without adequate documentation, programming team members will never catch on to what the job security specialist is doing. By attempting to obtain this information, the SAS programming manager risks severely annoying the job security specialist, but while this process can be painful, it is critically necessary if one hopes to disarm the job security specialist. Documentation is essential as it is the foundation of understanding exactly what the heck is going on. To help guide readers through any given program, all programs should include some type of header description as well as appropriate documentation throughout each section of the SAS program. For starters, a description of the program as a header to the program will give everyone a clear understanding of who, what, when, and how, etc. These fine details will allow one to easily identify and/or understand a given program. Below are typical examples of what one may see when comparing the job security specialist to a team player.

Job Security Specialist SAS Program Header

Program Name:	RICK57.ABC
Owned By:	Rick
Date Written:	2002
Purpose:	To be provided on a need to know basis.

Team Player SAS Program Header

Program Name:	Study_Enrollment_By_Site.SAS
Written By:	Ben Franklin
Date Written:	July 4, 1776
Purpose:	This program generates mean statistics comparing stars and rows on the flag.
Modifications By (#1):	Betsy Ross
Date Modified (#1):	July 11, 1776
Modification Description:	Additional parameters provided to accommodate more stars on the flag.

If one were to pick up the program knowing nothing at all about it, he or she should be able to review it and be able to answer some basic questions:

- What is the purpose of the program?
- What is the purpose of this DATA Step?
- What is the purpose of this PROC?
- What is the purpose of this macro?
- What is the purpose of this DO loop?

The bottom line is that programming managers should not approve code and deliverables until appropriate documentation has been included throughout the SAS program. Waiting until someone leaves the company or needs to transfer the work is pretty much too late. At any given point in time, anyone should be able to access a program, generate the output, and perhaps even know what to do if minor modifications are needed.

BRINGING THE JOB SECURITY SPECIALIST ONBOARD

Fast and Easy Tip #9

Ask the specialist to present a new or unfamiliar SAS PROC to the team.



Now that the job security specialist has been disarmed, it is now time to work on bringing this individual onboard the team train. After all, the job security specialist was of concern because he or she is in fact a programmer who probably can make significant contributions to the team. Proper channeling of this individual's efforts could result in a more positive outcome. A few possible approaches that could be followed to bring the job security specialist on board may include:

- Promoting the sharing of code
- Encouraging programmers to "show off their stuff"
- Promoting team participation
- Promoting user group participation
- Encouraging participation in SAS brown bag lunches
- Encourage programmers to write papers or simply provide informal documentation on neat techniques

Get the message across that the job security specialist's code is special, unique, and valuable to others. Go with the concept that every painting is a Picasso and every programmer may potentially benefit from the sharing of code, techniques, and output. If you are successful in bringing the job security specialist onboard, then your programming team will be stronger than ever!

CONCLUSION

Fast and Easy Tip #10

Smother the job security specialist with random acts of kindness!



SAS programming managers and coworkers no longer have to be fearful of the job security specialist. Approaches have been presented in this paper on ways to deal with the job security specialist with a positive emphasis on taking the direction of ultimately bringing this individual onboard with the rest of the team. Negative approaches are often destined for failure as team members scatter even farther apart. If presented to the team in the right manner, everyone may realize the benefits that are involved for all. Although a given team member may often shine for his or her high level efforts, there can sometimes be a fixed ceiling on just how high these efforts can go. By bringing everyone into the loop, there can be endless limits for what the team can achieve. If one team member wins, then the whole team wins. In closing, intentionally "annoying" the job security specialist is a behavior that generally results in negative consequences and annoyances should be avoided at all costs. However, "annoying" can be fun at times ...

REFERENCES

- Carpenter, Arthur L., 1996, "Programming For Job Security: Tips and techniques to Maximize Your Indispensability", presented at the 21st Annual SAS Users Group International Conference and published in the conference proceedings
- Carpenter, Art L., 1998, "Programming For Job Security Revisited: Even More Tips and Techniques to Maximize Your Indispensability, presented at the 23rd Annual SAS Users Group International (SUGI) Conference and published in the conference proceedings.
- Mitchell, R.M. (2001). Fast and Easy Way to Annoy a Statistician: The Sharing and Presentation of Data Between a SAS Programmer and a Statistician (Paper 243-26). *Proceedings of the 26th Annual SAS Users Group International (SUGI) Conference*.
- Mitchell, R.M. (2002). Fast and Easy Way to Annoy a SAS Programmer: A Statistician's Revenge! (Paper 256-27). *Proceedings of the 27th Annual SAS Users Group International (SUGI) Conference*.

- Mitchell, R.M. (2003). Fast and Easy Way to Advance on Your Beginning SAS Programmers! (Paper 8-28). *Proceedings of the 26th Annual SAS Users Group International (SUGI) Conference*.

ACKNOWLEDGEMENTS

SAS is a registered trademark of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are registered trademarks or trademarks of their respective companies.

DISCLAIMER: The contents of this paper are the work of the author and do not necessarily represent the opinions, recommendations, or practices of Westat.

CONTACT INFORMATION

Rick M. Mitchell
Westat
1650 Research Boulevard, WB 496
Rockville, MD 20850
(301) 251-4386 (voice)
(301) 738-8379 (fax)
RickMitchell@Westat.com