Paper 144-30

Litigation and SAS®: Some DOs and DON'Ts *(or, you call that Evidence?)*

Nicholson Warman, Peopleclick, Inc., Raleigh, NC

ABSTRACT

This paper will cover some steps to follow when preparing SAS programs and data results as evidence for litigation, be it for discovery and/or trial. Some common errors will also be covered, and steps to guard against these errors will be covered. With SAS used more and more extensively to analyze data from all industries, the data and programs need to be handled correctly, or your efforts could be suppressed under a Daubert motion. And you thought that company bonuses would be good this year!

INTRODUCTION

You and your company are suing (or being sued) by another organization or group of individuals and this is termed "the Complaint". What are some of the legal considerations (excluding guilt or innocence)? The key points are:

- Some brief concepts of e-Discovery
  - ✓ Federal Rules Of Civil Procedure
- Cooperative Compliance
  - ✓ v. Failure/Unwillingness to provide evidentiary data; Spoliation
- Timeliness
- Statistical Significance
- Reproducibility
- Time Box – a means to put together Time Series data in context
- Chronology
  - ✓ Continuity
- Program Structure
- Interactive v. Batch

Let me also make clear that I am not a lawyer. Your legal counsel is YOUR partner and KEY to success in the jurisdiction of the court system in which this case is to be heard. Different courts and/or jurisdictions have differing rules on what constitutes the period of data applicable to a case, and what measures have the greatest import to the court. An argument before the Supreme Court would be structured differently than a case in Federal Court or a more "junior" court. Further, since I work largely on Affirmative Action/Discrimination cases, the examples and discussion will tend to be from that arena, BUT the same principles would apply in a financial or other arena.

SOME BRIEF CONCEPTS OF E-DISCOVERY

There are a series of stages through which a case proceeds.

First, there is the Complaint. This document declares who is the Defendant and who is the party laying the Complaint (the Plaintiff). It also covers the period of the complaint (the Claim Period), such as January 13, 1995 to date. You will need this date, as this is the starting point for all data that must be retained and made available to the Court, should the case proceed. More on this later in Cooperative Compliance.

Once the Court has accepted the Complaint, you have a Case. Counsel will advise you the due date for providing data, and since time is of the essence, you will need to organize the data, be they databases, text files, PDF files and/or images. There are even companies today that will offer the service of taking your paper documents, scanning and indexing them, and making them available on the Internet in secured sites.

There also can be a series of exchanges as Opposing Counsel learns the data supplied and its interpretation. As in most court proceedings, before a case gets to trial, there is then a stage called discovery (where electronic data and electronic imaging are involved, the term is now e-Discovery). This is where both parties exchange the evidence they have. This is also an opportunity to depose (that is, interview) the witnesses for Opposing Counsel and to verify the evidence that they plan to present. Through this stage, both parties see the opposing side's general plan of attack, and the strength of the evidence being brought to bear. If there is an opportunity present, settlement talks can begin.

FEDERAL RULES OF CIVIL PROCEDURE

These Rules (and others particular to the Court jurisdiction for this case) are the basis for the introduction of evidence. They address what and when evidence may be included and/or excluded. These Rules drive damages should these Rule(s) be abrogated. They were last published December 1, 2001 as FEDERAL RULES OF CIVIL PROCEDURE a.k.a. "The Rules". Think of them as a system specification, giving "*what to do*" and "*what to not allow*", along with exception handling ("*In the case that … does not meet …, the punishment shall be …*"). They include the governance

of the Court and its general operating methods. The Rules are the means by which evidence is "discovered", accepted and presented to the Court. From The Rules has grown a large body of case law, which is another way of saying that there have been many cases that have given their interpretation of Rule _X_ or _Y_ as follows …. Arguments are made based on The Rules and this body of case law. The more Precedence (or more times that a similar ruling has been made in a favorable fashion to your case's evidence) giving support to your case, the stronger your arguments. What is typically found, on a large case, is your corporate Counsel will engage outside Counsel to actually manage the case and the court proceedings/evidence/etc.

You should further be aware that the US Courts system is in the process of making changes to The Rules, and in the Web References section, you will see a link to the (current) web page on this topic, allowing you to see the latest position and drafts of these amendments, particularly referencing e-Discovery. February 2005 is the deadline for comments to the proposed amendments.

COOPERATIVE COMPLIANCE

As a rule, when you are asked to provide information to a Court, you are expected to balance two objectives, namely "the Truth, the whole Truth and nothing but the Truth," and "only answer the question as posed." On the one hand, it means to provide EVERYTHING that would possibly bear on the case. On the other hand, it means to only supply those fields and records that are demonstrated to be essential to the case, leaving out everything else.

In e-Discovery, there are two extremes we will cover; a reasonable choice is somewhere between the two. On the one hand, "dump" the database with the data somewhere in it, and provide that as the response for data. The number of questions from Opposing Counsel would be reduced, as all the relevant information is found in the "dump." Since this is easily done, it has low costs for Counsel and client. This may mean that Opposing Counsel's costs are increased, as they have to examine everything to determine what is relevant. As an example, using a financial system, this could be a data dump of all transactions against the General Ledger, but without the client names, addresses, etc., just their account numbers; compare this to what was being sought, which was information on Sales Taxes levied.

On the other hand, you analyze your data holdings and only provide custom extracts that meet the letter of the complaint. The problems with this approach are first, the time and cost to create the extract, then the to-and-fro of questions and responses to explain the data. It also leave one open to the charge that only favorable data were extracted and that the "problem"/unfavorable situations had been removed. Reasonable doubt could be introduced because of this well-meaning action. You should anticipate that, the greater the degree of filtering of data, the greater the risk of a negative ruling in this case. A charge of tampering with the evidence due to excessive filtering to make it "meaningful/relevant" is a possibility if you excessively filter the data to "assist" Opposing Counsel in accessing your data. A balance between the risks and benefits of the two approaches is necessary to determine the least expensive method with the fewest risks to follow in complying with a Court order to provide this data. See the next section on Spoliation for further discussion.

As part of Cooperative Compliance, you should provide some notes on how the various extracted data tables relate to one to the other, and optionally, some information on the code sets (think `PROC FORMAT` for values and labels, `PROC FREQ` for code sets and value sets, counts for unique values) in the data tables supplied.

v. FAILURE/UNWILLINGNESS TO PROVIDE EVIDENTIARY DATA; SPOLIATION

Spoliation[1] is by legal definition "Intentional alteration or destruction of a document."

Failure or unwillingness to provide the data on the basis that the data were deleted or the system containing the data made it too difficult to extract the data can be grounds for the Court to censure your side of the case, up to and including summary judgment against you. If your information system is too complex to allow the ready extraction of the needed data, there are two possible outcomes:

- ✓ it is possible that the judge will order its extraction with part of the fees to be paid by the requesting party.
- ✓ if the system is so complex that the data still cannot be retrieved, this can lead to censure by the Court , up to and including summary judgment against you.

Be aware that when a Complaint is filed, it will indicate the Claim Period effective for that Complaint (such as January 13, 1995 to date). While a Court's jurisdiction may include rules that reduce this Claim Period (for example, a maximum of seven years in some jurisdictions prior to the date the Complaint was filed), expect to be able to present all relevant data for the period of the Complaint. If you are one of the unfortunate organizations in the news recently being hit with multiple complaints, the data required for each complaint may not be deleted, archived or otherwise modified until that particular case is resolved (and possibly not until all appeals are heard).

Know your data management practices and policies, as if it is corporate policy to automatically archive data after 90 days, summarize it after 180 days (deleting the original transactions), and delete the summary after 365 days, declare this as soon as you are notified of the Complaint. If this is only revealed at Discovery, sanctions may be imposed by

the Court, up to and including summary judgment against your organization, due to Spoliation. This will particularly apply if you have continued this destructive practice since the Complaint was received, as you are legally required to protect all the transactional data evidence required for the case while the case is active, beginning with the data from the start date in the Complaint. It all comes down to the intentions of the parties involved.

TIMELINESS

As can be seen from the following, quoted from daubertontheweb.com (Jan. 13, 2005 – federal appellate decisions within the past thirty days):

> "***Steppin' Out, Inc. v. National Sav. Corp.***, No. 01-17282 (9th Cir. Jan. 6, 2005) (unpublished). District court excludes testimony from defendants' expert in trial of copyright infringement action. ***Affirmed.*** Exclusion was proper discovery sanction for expert's submission of tardy and incomplete report."

While reasonableness applies, time is of the essence when producing data or evidence. Any delay that cannot be reasonably excused by the Court can lead to disqualification and could lead to summary judgment for the case.

STATISTICAL SIGNIFICANCE

Now that we have data to analyze, what would statistical evidence be without a discussion on statistical significance? Regardless of the statistical tests used, the number of Standard Deviations (SDs) or p-Values, they are the measures used to signify bias for/against the Defendants' claim. Expect that p-Values will need to be expressed in Normal-equivalent SDs for the Court; expressing them as a Probability may suffice, but you then have to discuss whether this is a one-tailed or two-tailed probability and in what distribution (e.g., binomial, two-tailed). The courts are expecting Normal Distribution SDs, although they are generally referred to just as Standard Deviations.

Most if not all Courts now accept that a minimum of two or three for the number of Standard Deviations (SDs) constitutes proof of bias. The stronger the rest of the evidence, the "weaker" this measure can be yet still support your argument. As an example, in a company with 100 divisions, a claim of age discrimination would need a number of divisions displaying in excess of -2.00 SDs for such a claim to be supported. If the SD calculation for the totality of the organization was also in excess of -2.00 SDs, then the claim would more likely "stick" as there is evidence of systemic discrimination/wrong-doing across the organization.

Imagine on the other hand that only two or three of the 100 divisions had negative numbers of SDs, but those SDs were about -15.00 SDs apiece. This would show that the problem is not systemic but rather is strongly localized to just those divisions. A claim against the company as a whole could be refuted, as a result. If the company, as a whole, had in excess of -2.00 SDs, then there would still be evidence of company-wide bias. Based on other evidence, Counsel should be able to use this statistical evidence to support their claim/defense.

Regardless of the statistical test used, so long as it was suitable and appropriate to the situation, the courts support this use and interpretation of standard deviations as evidence, all else being equal. Both Rank Sum and Regression tests are most frequently used in the human resources discrimination arena, however the nature of the data and the basis of the case would determine what other test(s) would be appropriate.

REPRODUCIBILITY

This speaks to the ability of Opposing Counsel to receive your programs and data sets, and reproduce your results, with only minimal effort on their part. Having to change file paths and drive letters is not a significant effort, and with proper documentation, should be easily achieved by them. If they CANNOT reproduce your results, however, given your code and your data, then the admissibility of your evidence and conclusions could be rejected by the Court. If you were a paid consultant providing this service, know that if your evidence is suppressed by the Court as being inadmissible (vis. a Daubert Motion[2]), your chances of being paid are reduced.

In order to make everything clear to Opposing Counsel (and thereby to the Court), reinvent the old concept of the Run Book. In the punch cards and mainframe days, when submitting a batch job to a data center, you had to specify what program was to be run after another program, the input files required for each and their respective location(s) on your network/server, and any outputs created. Likewise, for the sake of reproducibility, take the time to document the programs and data you are supplying, their location on your network, and where they are being stored on the transmission medium (such as CD5 \Folder5\mydata.sas7bdat). By providing this level of documentation, you will also discover quickly where included code was used, and be able to provide it to Opposing Counsel. Failure to supply this code leads to exchanges between Counsels to gain access to this code. And, if the code cannot be supplied or explained, then the result can be that all your efforts were for naught, and your conclusions discounted/suppressed by the Court, all because Opposing Counsel could not reproduce your results. The rationale is that if your work cannot be reproduced, then all conclusions and/or subsequent processing results that depend on the faulty program/data are questionable. Questionable data leads to reasonable doubt, one of the most loved and feared phrases in law, depending on how it affects your case.

To make the concept clearer, the following is a Sample Run Book:

Seq #	Program	Input	Output	CD# \ Folder
1	C:\myjobs\C1\gettext.sas	C:\myjobs\C1\clients text file	L:\contracts_1\Client Data\Raw_text.sas7bdat	CD4\Misc
2	S:\Contract_1\final programs\Read data.sas	L:\contracts\Contract_1\Client Data\Raw_text.sas7bdat	N:\mydata\contractContract_1\rawdata\Edited_text.sas7bdat	CD7\Input
3	S:\Contract_1\final programs\Read data2.sas	X:\contracts\Contract_1\Client Data\Raw_text.sas7bdat	X:\Hubert's client\Edited_text.sas7bdat	CD2\Study
4	S:\Contract_1\draft programs\Read data.sas	C:\Data\Raw_text.sas7bdat	C:\Edited_text.sas7bdat	CD1

Note that the input files for job 2 and 3 are the same file; only the drive letter used on the respective computers is different. Documentation provided with this Run Book indicated that the L: and X: drives are mapped to the same point, so that references to L: and X: are interchangeable. Further, the input file for job 4 is a copy of the file used by job 2 and 3.

By comparison, assume that only the CDs are supplied; no documentation is supplied (such as the preceding Sample Run Book). Considering only the above four jobs, expressing the results in Run Book format, you would see:

Seq #	Program	Input	Output
1	CD1\Read data.sas	CD1\Raw_text.sas7bdat	CD1\Edited_text.sas7bdat
2	CD2\Study \Read data2.sas	CD2\Study \Raw_text.sas7bdat	CD2\Study \Edited_text.sas7bdat
3	CD4\Misc\gettext.sas	CD4\Misc\clients text file	CD4\Misc \Raw_text.sas7bdat
4	CD7\Input\Read data.sas	CD7\Input \Raw_text.sas7bdat	CD7\Input \Edited_text.sas7bdat

You can quickly see the confusion that is created because of the copying process, and without a document map from the original file structure to the CD structure, the chaos that will ensue. Imagine the recipient's joy at receiving such a set of CDs with no documentation, and multiple copies of the same file name, each possibly with different creation date-times, locations, structures or contents! And, the attendant joy of trying to match multiple programs to multiple data sets, to get a result (any result!) that conformed to the results supplied. Note that without the originating Run Book information, the input files for job 2 and 4 are appearing as NOT being the same file! The recipient now has to find which program(s) spawned which of these files (or in reality, all of them, since they are just copies of the same file!). Worse still, you have no way to easily determine which program is to be run after what other program, at least not without a LOT of leg work and skull scratching. Just because the data and program files ended up on the same CD, you cannot assume that they are related, one to the other. Nevertheless, it IS a good starting point, barring better information.

NOTE: every CD would be given its own, unique Bates number, so that it is registered in evidence for the case.

This in no way speaks to whether your methodology and practices are correctly applied, just that both parties can get the same program to provide identical data, given the same input. The analysis that follows is to find weaknesses and/or faults in the logic used. However, that is a tale for another day.

TIME BOX – A MEANS TO PUT TOGETHER TIME SERIES DATA, IN CONTEXT

One of the hardest problems, when working with various sources of data, is putting them together into some coherent time series, so that the interactions over time can be seen more clearly, in their proper context. Most systems are transaction oriented, which means that they focus on the Effective Date, and the transaction has life until the next transaction. The problem comes into focus when you bring two files together, each transactional in nature, with sometimes differing dates for purportedly the same event. The analogy is the difference between the fence post and the fence rail; the rail (duration) exists between the two fence posts (transactions). Think:

← *Implied Event 0* *Event 1* →	*Event 2* →	*Event 3* →	*Event 4* →	*Event … n* →	…	← *last event*
period 1	*period 2*	*period 3*	*period 4*	*period … n*	…	*last period*

As an example, a database table that has descriptive text to explain various codes has records such as:

Code	Label	Effective Date
M	Male	1/1/1970
M	male	1/2/1970
M	Male/Hombre	1/3/1980

How do you match the appropriate record to your personnel files for a report? Do not do what one person did, which is dump the database into a block of text that was then copied (didn't use %INCLUDE!) into each program. The code looked like:

```
. . .
if SEX = "M" and TRANS_DATE >= '1Jan1970'd then DESCRIPT = "Male";
if SEX = "M" and TRANS_DATE >= '2Jan1970'd then DESCRIPT = "male";
if SEX = "M" and TRANS_DATE >= '3Jan1980'd then DESCRIPT = "Male/Hombre";
. . .
```

for each of the thousands of code records in their system! How the time box concept relates here is the recognition that these records are transactions (or fence posts), and we need them to be expressed as durations (or fence rails). This would make the data:

Code	Label	Effective Date	Closing Date
M	Male	1/1/1970	1/1/1970
M	male	1/2/1970	1/2/1980
M	Male	1/3/1980	1/1/5000

To get from the first to the second code table, you would first sort your data by Code and descending Effective Date. The Closing Date for the current record would be some date in the future that you will not need to worry about in your calculations. A missing value can be used, but it takes extra programming to resolve the time series in that case. I pick January 1, 5000 as being an unlikely date to naturally occur in my date ranges. Using FIRST. logic, the DATA step would look something like:

```
data WORK.CODE_DURATIONS;
     set WORK.CODE_TABLE;
         by CODE;
     retain CLOSING_DATE;
     format CLOSING_DATE  mmddyy10.;   /* pick the date format of your choice */
     if first.CODE then CLOSING_DATE = '1Jan5000'd;
     output;
                   /* next record closes day before our record starts */
     CLOSING_DATE = intnx('day',EFFECTIVE_DATE,-1);
     Run;
```

Be aware that some kind souls will provide data where the EFFECTIVE_DATE is the same for multiple records with the same CODE, causing a CLOSING_DATE to be the day before the EFFECTIVE_DATE, precluding its selection in any matching. Note too that because of multiple dates for the same code, we cannot use a PROC FORMAT to label our codes; PROC FORMAT requires unique values for it to look up the LABEL for each value.

Now that we have our durations, life cycle, fence rails, etc. determined, how would we match to some staff records?

Employee ID	Name	Sex	Hire Date	Terminated	Other fields in the table
1	Black, Joe	M	6/25/1975	12/31/1985	.
2	Doe, Jane	F	7/1/1973	.	.
3	Guy, New		1/1/2006	1/1/2006	.

To add our LABEL field to this file, we would probably use PROC SQL, since the MERGE statement would require our sorting our data on our VALUE fields, a potentially expensive proposition. While the limit on the length of this paper precludes the diagrams with this matching criteria, try playing computer with the WHERE clause, and you will see how the possible seven enumerations ARE correct. I have not included the logic of handling missing values here for ease of comprehension, but in reality, you would have need to control for missing values in your data.

```
proc sql;
     create table WORK.X as
            select a.*, b.LABEL
              from WORK.STAFF_RECORDS  a
                        left join
                   WORK.CODE_DURATIONS b
                on a.SEX = b.CODE
             where a.HIRE_DATE <= b.CLOSING_DATE
               and a.HIRE_DATE >= b.EFFECTIVE_DATE
             order by . . .;
     quit;
```

If we were matching two durations, rather than a duration to a transaction, the WHERE clause would read instead, a.START_DATE <= b.CLOSING_DATE and a.END_DATE >= b.EFFECTIVE_DATE. While this appears strange,

what we are doing is ensuring that the periods OVERLAP by at least a day. Understand that this can cause duplication of records. This is correct, for most situations!

As an example, let us look at Joe Black (employee #1), who was employed from 6/25/1975 to 12/31/1985. For every record coded with these dates on STAFF_RECORDS, we would have two records in the result set. This is due to the code table having two different meanings for the SEX value of M: "male" from 1/3/1970 to 1/2/1980, and "Male/Hombre" from 1/3/1980 forward. Do not assume you can just recode all the SEX='M' values as 'Male', as you have not respected the original data information integrity. While with some careful study, you may be able to state that, for purposes of your study, all SEX='M' are 'men', SEX='F' are 'women', and every other value is 'other'. If you do not make this declaration, your analysis may be judged to be fundamentally flawed, reducing its value as evidence.

I worked with one organization that had five sexes in their organization, and for some studies, all five sexes were analyzed independently. The sexes were: Male, Female, Candidate (gender neutral selection), Refuse to provide, and Not disclosed (did not fill in any gender box); the first four choices were actually on the hiring and staffing forms. This highlights the vulnerability the analyst faces in making assumptions about the data. To use an example from the accounting realm, imagine the chaos if an analyst assumed cost accounting rules for a financial system, and the actual method used was accrual finance. How would one handle the contra-accounts in that case?

CHRONOLOGY

Let me put some questions to you. Is it reasonable for:

- A .SAS program file to have a last modified date four years after the associated .LOG and .LST files?
- A .LOG file with a last modified date 20 minutes after its associated .LST file?
- A .LST file to exist without an associated .LOG file?
- A series of programs, to be run one after the other, to have execution dates that are in the opposite order?
- The last modified date of a SAS data set to be later than that of the programs that use that data set as input?

I would argue that the first and third bullets should NEVER occur. The rationale given for bullet three is that the SAS programs were run interactively, so there would be no .LOG file. What this says is that there is no evidence relating the program with any datasets/outputs it creates, UNLESS Reproducibility above confirms that the SAS program really did create this output. Evidentiary processing generally means batch processing of SAS programs, so that the last modified dates for the program, LOG and LST (and other output files such as SAS data sets, HTML, EXCEL, etc.) all have consonant dates. This does not preclude developing programs and results interactively, BUT if your evidence does not exist as files having relatable dates, your work can be challenged as having little to no value to the Court proceedings. This can seriously compromise any evidence based on this processing.

As for Bullet 2, realize that the last modified date (at least in the Windows environment) indicates when last the file was modified/had text added to it. It is not unreasonable, when working with large files, that 20 minutes of wall clock time could pass between the output of the last report and the completion of all activities, such as a `PROC SORT` of a large file. The information to explain this gap can also be derived from the LOG file, by examining the run time for the step(s) following that step which created the last output file.

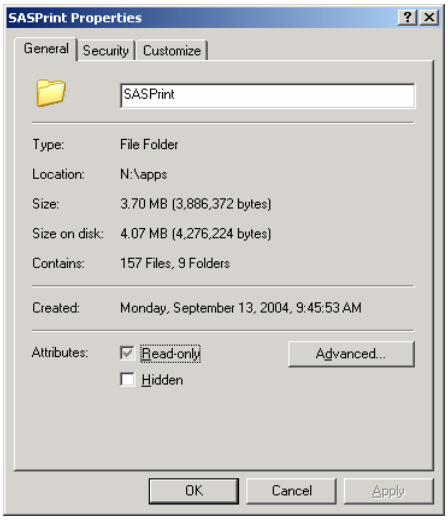
CONTINUITY

To expand on the issue of Chronology, and addressing bullets 4 and 5 above, assume that to produce the numbers entered as evidence takes 10 programs, moving from the initial (raw) data from the Plaintiff/Defendant, to the final listing. I would suggest that the last modified date for each program needs to conform to the run order, so that the earliest program in the process shows that it finished execution BEFORE any program(s) that depend on its output. Thinking about the SAS data sets for a moment (and bullet 5), how would one explain the situation where a program uses data *before these data were created?* Obviously, the early file actually used by the program has since been overwritten. So, how do you prove that the replacement file is identical in type, content and function with that it replaced? It is by happenstance, and not by "proof," should reproducibility be able to confirm the results.

As you can see, Chronology applies within a program and its components, but also must respect the Continuity of earliest creation to latest use. If you cannot get the timeline right, then you leave yourself open to a challenge on the validity and reproducibility of the numbers supporting your presented opinion entered into evidence.

To paraphrase, "if the dates don't fit, you must acquit!"[3]

A way to avoid the chronology and continuity problems, again in the Windows environment, is to go to the Windows Explorer. Select the .SAS, .LOG and .LST files, and the associated output file(s), and right-click to get File Properties. Select the READ ONLY property and click on OK. This means that should someone try to rerun this program, it will fail without damage to your .LOG, .LST, .SAS7BDAT and other output files. This simple trick will save you hours, in attempting to recover from mistakes, and serves as a warning to others to not tamper with these files, as they are part of your evidence set. Wow! A simple solution! Who says that this has to be complicated or difficult!



Screen shot from Windows/XP

PROGRAM STRUCTURE

It constantly amazes me in this day and age that people are still allergic to white space, comments and `RUN;/QUIT;` statements!

A well-written program should have a set of (sometimes-abbreviated) comments to indicate the purpose of the program, and any information such as the context and assumptions about the data and/or process. This allows the reader to know what program X.SAS does. Ideally, the program name would provide that information, but on some computer architectures, the program name is more likely required to be something as mnemonic as S01P001I. The comments should also show at a minimum the initials of the author. There should also be a "maintenance log" set of comments. These would detail changes/developments in the program from initial conception to final use. Here you find needed information on such things as coding errors in the data, cautionary notes on merges and joins, etc. that are not otherwise readily available.

A well-written program has clearly defined "units of work," which begins with a `DATA` or `PROC` statement and ends with a `RUN;` or `QUIT;` statement, as appropriate. There should be only one statement on a line, but a statement such as `DO`…`END` or `IF`...`THEN`...`ELSE`... should have appropriate indenting to make the logic clear. Global statements should appear immediately preceding the procedure or `DATA` step to which they apply. The `OPTIONS` statement should be the first occurring statement in the program, followed by the comments such as program name, maintenance log, etc. As an example, but not the only way to write the code:

```
data WORK.X;
     set MYLIB.DATA;
     if X gt 40
        then AGE_CLASS = '> 40';
        else AGE_CLASS = '1-40';
     run;
title2 '... ';
proc print data=WORK.X;
     ...
```

and not

```
data X; set MYLIB.DATA; if X gt 40 then AGE_CLASS = "> 40"; else AGE_CLASS = ↵
"1-40"; proc print data=X; title2 "..."; ...
```

which works syntactically but is miserable to read, especially when the line reaches over 250 characters long. As an aside, while interactive SAS permits any line length you wish to type, batch SAS is limited (at least on the Windows platform) to a line length of less than 255. Therefore, programs in the second format will cause errors, and we are back to Reproducibility issues again. Further, the second approach to program coding indicates a program written in haste, and experience says that this is the first place to look for mistakes/problems. It displays a programming style referred to by a university colleague as "doctoral donkey twaddle." This refers to their PhD students who were only interested in the results of the class assignment, and the program(s) needed to reach those results were viewed as just a "means to an end." Such programs are most typically the source of forgotten inclusions, randomly created

variables in the datasets, and assorted other errors/omissions/etc. Some of the best evidence against the admissibility of Opposing Counsel's evidence can be found in such programs, even though these programs are a pain to read and understand.

This discussion of readability reaches, in my experience, near-religious fervor for many. While the process may be clear to you, as the author, most recipients will appreciate the assistance of comments, white space, clear units-of-work, etc. Further, a .LOG file provides the exact expansion of the .SAS program and the information on execution, including fully qualified file and library path names. Trying to knit together what was done, when the source program's data path was given as "..\myfile\sasdata.sas7bdat" does not make it possible to determine which of several directory folders called \myfile\ were being referenced. The system expansion to the fully qualified path and name, on the other hand, makes it clear what file path is being used, and exactly which program is using specifically which file. This ability to relate like with like will prove critical when reproducing the original analyst's results.

My recommendation is to provide sufficient comments and in sufficient detail, that someone not versed in your operating environment can determine which file goes with which program in a coherent manner, and not just depending on pure, blind luck to marry the various components together as a whole. Documentation that you would include for a production environment indicates the type and amount of information you should provide in your evidentiary programs, so that Opposing Counsel can understand what they have received.

INTERACTIVE V. BATCH

This debate goes back to "ease of development" versus "reproducibility." While it is far easier to just type, letting the SAS Enhanced Editor reveal your errors in syntax, it also leads to a "stringy" and difficult to read style of program. Worse still, it is a never-ending source of errors/missing code, as you can add and remove code that was executed and influenced the final dataset output, but no record of what was ACTUALLY executed exists. How would you deal with the following?

A dataset is supplied in evidence with the following fields and code sets,

ID	Age	Sex	Y
1	19	M	•
2	25	F	•
3	52	F	•

where AGE is 18 through 65 or 99 for retired, SEX is M or F, and Y is always *missing*? This could arise from program code (see the following example), where, in one of the several runs interactively, someone introduced this extraneous variable. Since the executions were interactive, no history (that is, a .LOG file) exists to document what the logic used and how it was related to the case! And with the code removed from the .SAS file, we have no information on this variable, not even its purpose (if it has any!).

```
data WORK.X;
     set WORK.X        (obs=0)
         WORK.NEW_DATA;
     run;
```

To ensure that the variables and any labels created (for which we may not have a source program!) are retained and to ensure that the file will have the base variables for subsequent analysis, the `OBS=0` dataset option is used. This approach ALSO propagates temporary variables which are not a normal part of the processing but which occurred interactively in error. The further problem with this approach is that there is no documentation for the labels and variables that appear in the dataset. This is an example of what you should *never* do, as it prevents the program from being successfully executed in Reproducibility. Understand that you have now created a pre-condition that the file **WORK.X** must exist *before* this program can be executed! Isn't doing it correctly the first time easier?

Another favorite problem is where the above variable Y actually has data (such as "Over" or "Under"), but no code exists in evidence to define how the variable Y was created and no information is available as to which conditions caused this valuation for this variable. Neither case is reproducible without assistance from the originating party! If this variable is never used in subsequent processing, then it does not substantively weaken the evidence. However, if it is used elsewhere, it also puts into question whether the subsequent processing that is dependant on this dataset *may be permitted to be introduced as evidence*.

If batch processing creates and documents the datasets, then the resulting analyses are not suspect (due to their origin) and these questions do not arise; everything is available for inspection. Even if not all of the program code is in the SAS file, then it may appear in the LOG file. On this basis, there are three choices:

- make a request for the included file through your Counsel,

or, if the LOG is complete,

- copy/paste and edit the LOG material for inclusion in the SAS file, or
- retype what was included in the LOG file.

All of these approaches allow you to reproduce the results provided, and should deliver the same results. If you do not arrive at the same result, then either you do not have the correct input dataset(s) OR the program provided was not the program used to create the output file.

CONCLUSION

Batch processing, Chronology, Context, Continuity and Cooperative Compliance are the keys to success. The Time Box can be your friend in correctly analyzing time series data. You need to deal with organizing and documenting the data provided to Opposing Counsel. With suitable care and attention to details, you can emerge a winner, and may even be able to help Counsel suppress Opposing Counsel's evidence, since they were not aware of (and did not follow) these ideas. Bonuses could be good this year because of this case!

TERMINOLOGY

While the definitions are my own, it helps to know the terms, as a sizable case will get very confusing otherwise.

- ✓ Complaint — when suing someone, or an organization, the formal declaration that there are grounds to sue that party are contained in the Complaint, a formal document presented to the Court with jurisdiction to hear the case. While normally informal discussions may occur before this point, this is the actual starting point to the lawsuit. Once the Court has accepted the Complaint, then the process begins. The Complaint also specifies the Claim Period, the period for which you need to provide data.
- ✓ Defense Counsel — this is the legal team defending the party being sued. Their clients are referred to as the Defendant(s), and against whom the Complaint was issued.
- ✓ Evidence — those items and/or facts presented in Court that demonstrate the guilt and/or innocence of the Defendant(s).
- ✓ E-Discovery — this refers to the discovery phase and is particular to electronic evidence, in all its forms. It is not just extracts from a database but it also includes the conversion of paper documents to computer images *in a controlled fashion, protecting the chain of evidence*.
- ✓ Litigants — this is the group of Counsel for the Prosecution and Defense as well as their clients and witnesses. Litigants appear before the Judge in Court and/or as Witnesses.
- ✓ Opposing Counsel — this is the legal team who oppose your team (OC for Defense is the Prosecution; OC for the Prosecution is the Defense).
- ✓ Plaintiff's Counsel — this is the legal team who will prosecute the case against the Defendant(s). They represent the person(s) and/or organization(s) who raised the Complaint. They are referred to as the Prosecution, and their client(s) are the Plaintiffs.
- ✓ Standard Deviation (SD) — a statistical measure that demonstrates the tendency of the data to vary from the mean. The larger the SD, the stronger the demonstration of a variance, with a minimum of 2 to 3 SDs being the cut-point for most Courts as evidence that there are one or more non-random events influencing the outcome of the test being measured. In the range of -2 to 2 SDs, random chance could explain the results; outside this range indicates deliberate action/omission of action. Remember that these are Normal Standard Deviations; for other distributions, you must convert them to Normal-equivalent Standard Deviations.
- ✓ Witnesses — those persons who appear in Court to give evidence pertaining to this case. Through their testimony, evidence is corroborated or refuted. They can be either expository witnesses (actually witnessed some or all of the events of the case), or expert witnesses (called because they can interpret details and events for the Court from a position of expertise – they did not witness the events of the case). Statistical evidence is typically presented and/or refuted by Expert Witness.

REFERENCES

Federal Rules of Civil Procedure *see also* <http://www.house.gov/judiciary> under print shop
House, Committee of the Judiciary
U.S. Government Printing Office
Mail Stop: SSOP
Washington DC 20402-9328
December 31, 2004
pp. 127

WEB REFERENCES

Peter Nordberg, 2001-2005
<http://Daubertontheweb.com> — a more in-depth discussion than time permits in this presentation.

Cable News Network LP, LLLP, 2005 (A Time Warner Company)
<http://www.cnn.com> — this is the on-line site for copy aired and/or printed on various topics.

WEBNOX Corp., 2000-2004
<http://www.hyperdictionary.com/dictionary/spoliation> — this is a web-based dictionary that includes computing and legal terms. Replace spoliation with the word sought in order to see the definition through this web site.

Law and Contemporary Problems
Duke University School of Law, 2001
<http://www.law.duke.edu/journals/lcp/articles/lcp64dAutumn2001p213.htm> — OF CHERRIES, FUDGE, AND ONIONS: SCIENCE AND ITS COURTROOM PERVERSION, by David W. Peterson and John M. Conley — it is all about the evidence, Daubert motions and p-Values. This is fun reading with a serious message about the misuse of statistics in evidence.

Journal of Forensic Economics
Published by the National Association of Forensic Economics
<http://www.nafe.net>

Secretary of the Committee on Rules of Practice and Procedure
Administrative Office of the United States Courts
Washington DC 20544
<http://www.uscourts.gov/rules/e-discovery.html> — the most up to date reference to the revisions under discussion on the Rules of Evidence pertaining to Discovery (and e-Discovery).

ACKNOWLEDGMENTS

Mr. Peter Skalak (Sr. Director) to making this opportunity possible.

CONTACT INFORMATION

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

Nicholson Warman
Peopleclick Research Institute
Peopleclick, Inc.
Two Hannover Square, 7th floor
Raleigh NC 27601
Work Phone: (919) 645-3674
Email: Nick.Warman@peopleclick.com
Web: http://www.peopleclick.com

[1] See <http://www.HyperDictionary.com/dictionary/spoliation> for a definition of the term. Change Spoliation for any term to look up that definition.

[2] Cited as *Daubert v. Merrell Dow Pharmaceuticals (92-102), 509 U.S. 579 (1993)*, it lead to over ten years of discussion in over 650 appellate decisions on what and when scientific (including statistical) evidence can be accepted. Colloquially, it means a motion to suppress evidence that is deemed to be scientifically [but in this context, statistically] invalid. For a more formal discussion, see the discussion at daubertontheweb.com.

[3] See <http://www.cnn.com/US/OJ/daily/9-27/8pm/> for the original press coverage of Johnnie Cochrane's statement, *"if it [the cap or glove] doesn't fit, you must acquit"* in the OJ Simpson trial.