SAS® GLOBAL FORUM 2016

IMAGINE. CREATE. INNOVATE.

What to Expect When You Need to Make a Data Delivery. . .

Helpful Tips and Techniques

#SASGF

What to Expect When You Need to Make a Data Delivery. . . Helpful Tips and Techniques

Louise Hadden, Abt Associates Inc.

QUESTIONS YOU SHOULD ASK REGARDING THE PROJECT

- Is there any information regarding data processing in the proposal / contract / grant application – or the RFP?
- Is there a DUA / DSP / DMP / AP in place for the project that would give you information as to the source of the incoming data, the security level of the incoming data, where the data is housed, the timing of the incoming data (waves, years, quarters, months, etc.), and/or any analytic activities planned for incoming data?
- Is there any information on the source of the data? i.e. is the data coming from a survey, web or otherwise? Claims data bases? Collected from public use files on the web? Are you creating the data? If so, how?
- Is there any documentation for the incoming data (we assume you will use SAS tools to document any data you create!)? Any usage notes that might be relevant?
- Are there any relevant company / government / prime contractor regulations that apply to the use of incoming and/or created data, i.e. HIPAA, FISMA, FDA, etc.?
- Is your incoming / created data subject to IRB regulations?
- Is there a data manager / data lead for the project (is it you?)
- Who else is working on the project? Anyone else with data processing and analysis responsibilities?

TIP: Keep copies of documents such as DMP, AP, SOW in an easily accessible folder

CONGRATULATIONS!

You're expected to make a data delivery! This poster will walk you through the questions you should ask, the resources you should check, the resources you should create, and the SAS® tools that will help you along the way. In essence, we'll travel back to the future to find out exactly what you need to be doing from the very start of your project, so you don't run "outtatime".

Drive on to Page 2 for Helpful Tips

Acronym Definitions:

DUA = Data Usage Agreement
DSP = Data Security Plan
DMP = Data Management Plan
AP = Analysis Plan
RAF = Restricted Access File
PUF = Public Use File
PHI = Personal Health Information
PII = Personally Identified Information
SOW = Statement of Work
IRB = Institutional Review Board

QUESTIONS YOU SHOULD ASK REGARDING THE DATA DELIVERY

- Is there any information regarding the data delivery in the proposal / contract / grant application – or the RFP?
- Is there PHI/PII in the data delivery? How is it being protected?
- Will you be creating RAF and/or PUF files?
- In what format will the data be delivered? SAS, Stata, delimited, MS Excel, MS Access, XML, relational data bases?
- What versions of software are required to be used?
- What platform will the data be coming from and be used on by the recipient?
- How is the data expected to be delivered? SFTP? Encrypted drives (if so, what encryption software?)
- Will code / programs be delivered to the client? If so, in which language(s)?
- Will associated files (format libraries, macro libraries) be delivered to the client? If so, in what form?
- Will documentation be delivered to the client? If so, in what format?
- What naming conventions should be used for files in the data delivery?

TIP: Keep any emails / memos / specs regarding the data delivery in an easily accessible folder

What to Expect When You Need to Make a Data Delivery. . . Helpful Tips and Techniques

PLAN FOR THE END OF YOUR PROJECT

Think about your project's close-out (and data delivery) BEFORE you think about its start-up. Looking at the products that you need to deliver first will enable you to build a data management structure to help ensure success.

- What are the desired outcomes of your project? Will there be a data delivery or a series of data deliveries? If there isn't written verification of that, you need to research all available materials (see page 1 and Getting Started) and record and store the information that you find.
- At the conclusion of your project, will your data need to be destroyed? Delivered? Kept? If kept, for how long, in what form, and where? Will back-ups including your data need to be destroyed? In what time frame?
- Think about the cycle of deliverables. Will your project deliver data on a daily / weekly / monthly / quarterly / yearly basis? Or will it be a one-time delivery? Will there be interim and final deliveries? What are the exact dates or date ranges associated with each data delivery?
- Does your company or client have any official close-out practices or documents that need to be completed?
- Does your company or client have any quality assurance practices for deliverables that need to be planned for?
- Who, What, Where, How, When are all important items to know at the outset.

Answering these questions will help you get started, inform your data management plan, and help ensure success with your data deliverable.

It is also important to consider HOW you will deliver your data. Your client may have specific requirements in terms of:

- File formats (SAS data set [what version? 32/64 bit? What platform? Xport/cport/native?], "flat" file, delimited file [space? Tab? Comma? Pipe?], MS Excel (compressed? w/macros?], MS Access (compressed?) XML [with a map or schema?], relational data base, etc.)
- Encryption (no, yes, if yes, what program / level)
- Transfer method (SFTP? External drive?)

DATA STORAGE CONSIDERATIONS

It is vital to store files in an organized way. Keep all programs, logs and output, source data and final data used to create deliverables in separate folders. Data folders containing PHI/PII should be clearly labeled as such. Have a clearly-defined plan for naming datasets, programs and variables. Create a deliverable folder to hold final data sets, documentation, and programs (if applicable).

DOCUMENTATION

Document processes with a process log: at a minimum, include process / program name, user, date and any relevant notes. Inputs and outputs with number of observations are also helpful.

Maintain a catalog of incoming and outgoing data files: at a minimum, include filename, user, to/from locations, and transfer method.

Use SAS® to create data dictionaries for original and analytic files, write auxiliary files such as format assignment code, make entries into a process log, and put documentation within program code.

GETTING STARTED

If there is not a data manager assigned to your project, and there is not a data management plan, that is the first place to start. Every project involving data processing and/or data deliveries, large or small, will benefit from a comprehensive data management plan. The DMP should cover guidelines and standards for:

- Storage practices and folder structures. There may be multiple platforms and multiple time periods (as well as multiple users) on your project.
- File and program naming conventions (see Data Storage Considerations)
- Documentation procedures
- Communication within the project team
- Input / Output cataloging
- Processing logging
- Peer review / quality assurance

The DMP should also include references to:

- Data security plan (DSP) if relevant
- Data usage agreement(s) (DUAs) if relevant

And incorporate:

- Analysis Plan (AP)
- Schedule of deliverables
- Delivery method
- Plan for project close-out, including data retention and destruction

If a DSP is relevant but not written, the data manager should ensure one is provided as a companion to the data management plan. In addition, the data manager should ensure that DUA(s) are properly executed and updated in a timely manner.

Drive on to Page 3 for Helpful SAS Tips

What to Expect When You Need to Make a Data Delivery. . . Helpful Tips and Techniques

USE SAS TO SELF-DOCUMENT

Use system macros and functions to document your logs, lists, output and SAS output: SYSFUNC(GETOPTION (SYSIN)) returns the path and name of the program; &SASDATE returns the date the program began; &SYSTIME returns the time the program began and &SYSUSERID returns the user ID of the programmer who submitted the job. These are available to all SAS users, and can be used in titles, footnotes, label statements, description statements, created variables and even a stored macro to populate program headers (see Session 8300, Glass and Hadden).

Build a process log using SAS, using the %WINDOW and %DISPLAY commands to solicit input from users. If the routine is included in programs, information from each run of each program will be collected including the information described above and more. (see Session 8300, Glass and Hadden). The program outputs information into an Excel workbook which is read in and exported iteratively so that any additional information entered into the workbook is retained.

If you receive data sets that do not have data set or variable labels, etc., use PROC DATASETS commands to modify the data set. This is far more efficient than saving a new copy of the data set (or [gasp] overwriting your data set).

Similarly, if you have unlabeled macros, graphics catalogs or format catalogs, you can use PROC CATALOG to "describe" them after the fact. (You can use the DES option when creating macro and graphics catalogs, but must use PROC CATALOG to modify format catalog descriptions.)

If you need to deliver formats specific to a project or a portion of a project, save your formats with a two level catalog name. For example:

```
PROC FORMAT LIBRARY = LIBRARY.Compendium2011Res;
   VALUE YESNODK 1 = "Yes" 2 = "No"  8 = "Don't Know";
RUN;
```

USE SAS TO MANAGE DIRECTORIES

Create a text directory listing of a project directory starting from the top level from a command prompt or through the X command in SAS including subdirectories (i.e. dir *.* /s > dirlist.txt)

Use as input to a SAS program (readdirlist.sas, available from the author) which uses SAS functions to parse each line in the directory listing, converting the information into useful variables, and exports the information into an Excel spreadsheet. This spreadsheet (or the originating SAS data set) can be used to locate duplicate file names, large files, etc.

directory	name	date	time	ampm	filetype	size	bytesize
S:\Projects\NH-COMPARE\Data_From_C1FromCMS		12/11/2014	02:10	PM	Directory		
S:\Projects\NH-COMPARE\Data_From_C2ReferenceFiles		1/12/2015	09:40	AM	Directory		
S:\Projects\NH-COMPARE\Data_From_C3GenRatings		1/6/2015	08:56	AM	Directory		
S:\Projects\NH-COMPARE\Data_From_C4OtherProcessing		1/6/2015	01:39	PM	Directory		
S:\Projects\NH-COMPARE\Data_From_C5Output		1/5/2015	04:43	PM	Directory		
S:\Projects\NH-COMPARE\Data_From_Cfoo.xlsx		1/5/2015	12:03	PM	Excel	14474	KB+
S:\Projects\NH-COMPARE\Data_From_CINSPECT_HARDCODE_20150101.txt		1/5/2015	12:03	PM	Text file	13425	KB+
S:\Projects\NH-COMPARE\Data_From_Cjan2015filelist		1/15/2015	11:53	AM	UNKNOW	0	0-B+

CONCLUSION

Preparing for a data delivery is a complicated endeavor. By going back to the future, and with the help of SAS tools, you can plan for a successful transfer of data.

Full code for SAS tips described on this page available from the author upon request.

ACKNOWLEDGEMENTS

The author gratefully acknowledges the contributions of her colleagues Daniel Gubits, Ryan Kling, Elizabeth Axelrod, Andreas Maier, Christianna Williams and Roberta Glass.

CONTACT ME!

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

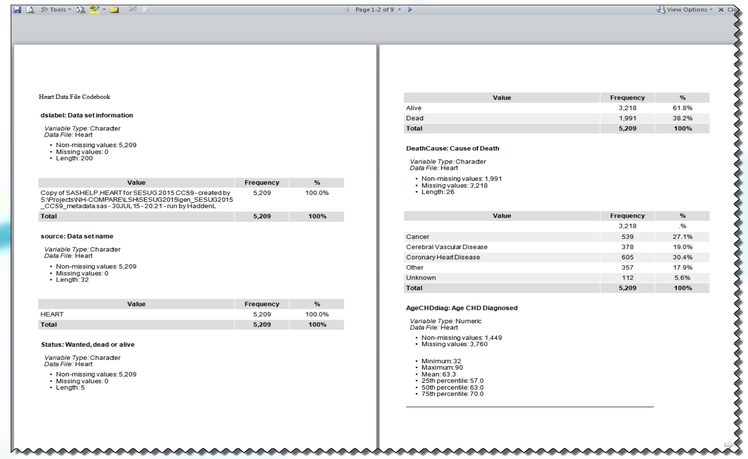
Name: Louise Hadden
E-mail: Louise_Hadden@abtassoc.com

Scan Me

USE SAS TO PRODUCE CODEBOOKS

Using all the tips described in this poster (and more), you are ready to document your deliverables. It is possible to produce customized codebooks for SAS data sets with very little user intervention.

- Use PROC DATASETS, PROC CONTENTS, or dictionary tables, etc. to produce a documentation spreadsheet
- Review the spreadsheet and perhaps modify a missing label or format assignment. In addition, you may wish to categorize your variables beyond numeric and character.
- Import the modified spreadsheet, and use the information to write code to be included to generate a codebook with output varying by variable type; write code to generate a label statement; and write code to generate a format assignment statement, among other normally onerous tasks.



Heart Data File Codebook

dslabel: Data set information

Variable Type: Character
Data File: Heart

- Non-missing values: 5,209
- Missing values: 0
- Length: 200

Value	Frequency	%
Copy of SASHELP.HEART for SESUG 2015 CC59 - created by S:\Projects\NH-COMPARE\LSHSESUG2015\gen_SESUG2015_CC59_metadata.sas - 30JUL15 - 20:21 - run by HaddenL	5,209	100.0%
Total	**5,209**	**100%**

source: Data set name

Variable Type: Character
Data File: Heart

- Non-missing values: 5,209
- Missing values: 0
- Length: 32

Value	Frequency	%
HEART	5,209	100.0%
Total	**5,209**	**100%**

Status: Wanted, dead or alive

Variable Type: Character
Data File: Heart

- Non-missing values: 5,209
- Missing values: 0
- Length: 5

Value	Frequency	%
Alive	3,218	61.8%
Dead	1,991	38.2%
Total	**5,209**	**100%**

DeathCause: Cause of Death

Variable Type: Character
Data File: Heart

- Non-missing values: 1,991
- Missing values: 3,218
- Length: 26

Value	Frequency	%
	3,218	%
Cancer	539	27.1%
Cerebral Vascular Disease	378	19.0%
Coronary Heat Disease	605	30.4%
Other	357	17.9%
Unknown	112	5.6%
Total	**5,209**	**100%**

AgeCHDdiag: Age CHD Diagnosed

Variable Type: Numeric
Data File: Heart

- Non-missing values: 1,449
- Missing values: 3,760

- Minimum: 32
- Maximum: 90
- Mean: 63.3
- 25th percentile: 57.0
- 50th percentile: 63.0
- 75th percentile: 70.0

Codebook generated by gen_codebook_SGF2016_Session8300.sas – full code supplied in proceedings and available upon request.

SAS® GLOBAL FORUM 2016

IMAGINE. CREATE. INNOVATE.

LAS VEGAS | APRIL 18-21

#SASGF