MANAGING LONGITUDINAL RESEARCH STUDIES:

PREPARING MASTER FILES

By

Linda L Remy, MSW PhD

Ted Clay, MS

UCSF Family Health Outcomes Project
Geraldine Oliva, MD MPH, Director
Jennifer Rienks, PhD, Associate Director
Linda L Remy, MSW PhD, Research Director

500 Parnassus Ave. Room MU-337
San Francisco, California 94143-0900
Phone: 415-476-5283
Fax: 415-476-6051
Web: https://fhop.ucsf.edu/

November 2018

TABLE OF CONTENTS

TABLE OF TABLES

TABLE OF FIGURES

TABLE OF LEGENDS

◌	From program creating incoming file or name of next program using file
⬮	Excel file input or output
⛁	SAS file input or output
🗎	Flat or text file input or output
PROGRAM NAME	SAS program with brief description of steps.

Suggested Citation

Remy L, Clay T. (2018) Managing Longitudinal Research Studies: Preparing Master Files. San Francisco, CA: University of California, San Francisco, Family Health Outcomes Project. Available at: http://fhop.ucsf.edu/data-management-methods

ACRONYMS

ASC	Ambulatory Surgery Center
BC	Birth certificate file
CPT	Current Procedural Technology
DRG	Diagnosis Related Group
DT	Death certificate file
DX	Diagnosis
E-Code	External cause of injury code
ED	Emergency Department
FHOP	Family Health Outcomes Project
HADR	Hospital Annual Disclosure Report
ICD-9	International Classification of Diseases, 9th Revision, Clinical Modification
ICD-10	International Classification of Diseases, 10th Revision, Clinical Modification
LOG	File SAS makes to report results of an executed program
LST	File SAS makes to display results of an executed program
MS-DRG	Medicare Severity Diagnosis Related Groups
OSHPD	Office of Statewide Health Planning and Development
PD or PDD	Patient Discharge (Data)
PX	Procedure
USPS	United States Postal Service
UCSF	University of California, San Francisco
ZIP	ZIP-code number assigned by USPS for mail delivery.

PREPARING MASTER FILES

This is the third in a series of documents describing basic methods the Family Health Outcomes Project (FHOP) uses to manage its longitudinal research studies [1]. Earlier documents describe how to set up the operating environment, and discuss basic standards to handle variables longitudinally [2,3]. Analysts working in local health jurisdictions and researchers interested in longitudinal research may find these helpful.

This document introduces specific methods to maintain longitudinal integrity of master files, within and across datasets. We show how to incorporate this methodology into SAS programs. After converting newly received population health files into the FHOP structure, we run a series of standard programs to review longitudinal continuity.

We are making this basic methodology and its associated software public to help population health researchers understand the nature of data management for complex longitudinal research. This also should provide a background to users of our longitudinal DataBook and EpiHosp products. We hope this will help people better understand how we preprocess master files to make these products and do our longitudinal research studies.

These methods produce master files whose contents are consistent within and across datasets, over time, and address source file idiosyncrasies and changes in content. All work is in SAS, assisted by Microsoft Excel and Visio. To replicate the process we describe here, download the file TOOLS.ZIP [1]. It contains the macros we discuss in this document.

OVERVIEW

Programs that read original, unencrypted master files into SAS contain macros that control what happens to any year of a given data source. An Excel spreadsheet defines variable names, length, type, position, labels, and formats. SAS code or macros incorporated into the Excel file tell SAS what it must do to make the file.

Each SAS program creating a master file has a simple prefix: e.g., patient discharge (PD), birth certificates (BC), death certificates (DT), emergency department (ED), ambulatory surgery center (ASC), etc. The program outputs 1 to *n* datasets depending on structure and content. The resulting program log and listing is composed of the prefix and year the data represents (PD2005). Dataset names also have a numeric suffix identifying the year (MAIN2005).

In this document, we provide an overview of our basic coding conventions, then describe how we receive and protect master files, read population master files into SAS, and check results for longitudinal consistency. For our example, we use the PDD, one of our more complex datasets. We show relevant SAS code from PD.SAS and include an example of an input Excel file that controls the process (PDV.XLS). After making master files, we run another sequence of SAS programs to understand longitudinal consistency of the data. The structure of ED and ASC datasets is similar to the PDD, all distributed by the Office of Statewide Health Planning and Development (OSHPD). Vital Statistics datasets follow a similar process but have different internal structure.

CODING CONVENTIONS

In reading our SAS code, observe that FHOP programs incorporate extensive internal documentation. Every program starts with a summary of the purpose, the major steps, input files, output files, program source, and programmer. This documentation is stored in several files that together create a longitudinal trail of the steps to complete a given study.

Our program logic often is complex. We typically start by writing what we are trying to do in complete sentences. As we write SAS code, we convert sentences to internal documentation, using a natural language style, with complete sentences and mixed case. The "hard knocks school" taught us that good internal documentation goes far to increase understanding of what we do in a program sequence. On many occasions, strong documentation has been vital to reconstruct our thoughts when we have had to modify programs written long ago. We rely heavily on Microsoft Visio to figure out steps in a program, and data flow across programs.

As much as possible, we avoid the use of slash-star (/*blah blah */) to delineate commenting. This can cause significant problems internal to macros. Starting a comment with an asterisk (*) and ending with a semicolon (;) avoids this problem (* blah blah ;).

We organize related tasks sequentially. We are not afraid to use space and try to limit line length to about 80 columns. Returns and spaces increase readability. Visually consistent indenting also greatly improves readability. These usually indicate changes in logic or task. We use long text blocks to delineate major steps, as in the following example.

```
*--------------------------------------------------------------------------*
* Identify California resident admissions                                  *
*--------------------------------------------------------------------------*;
```

We strive to use uppercase names for permanent variables and lowercase for temporary variables. Additionally, permanent variables are labeled and categorical variables formatted. It will be obvious in PROC CONTENTS that we forgot to delete a temporary variable or label a permanent variable. We prefer shorter over longer names for programs, files, and variables.

SETUP ACTIVITIES

Receive confidential master files

We receive confidential master files (hospital, birth, death, fetal death, etc.) zipped and password encrypted according to the rules of the agency sending the data. We copy the original file to an external drive and directory reserved for incoming confidential data (Confidential Drive) for example X:\PDD\RAW. We copy file documentation to the drive where we plan to store the standardized data (Master Drive), for example E:\PDD\SRC\yyyy, where yyyy is the year.

We move the newly received file into a ZIP file with a standard name, protected with a password FHOP developed. For master ZIP files, we use the 256-Bit Advanced Encryption Standard, which the National Institute of Standards adopted as the nation's Federal Information Processing Standard [4]. We eject the source CD, with its original password, and store it in a secure location.

All drives (internal or external) are password encrypted at the disk partition level using Microsoft BitLocker [5]. When someone connects the Confidential Drive to a computer, it will not show unless the user has the encryption software on the computer and knows the password. Settings make it impossible for any external drive to serve as a bootup source. When not in use, we disconnect the Confidential Drive from the computer and store it in a secure place. To protect confidentiality further, the Confidential Drive connects only to stand-alone computers. By our research protocol, only two members of FHOP’s team can access these files.

In addition to the stand-alone external drive with confidential files, we maintain a backup on a much larger external drive, also encrypted. Once a month, LR takes the large backup to FHOP on the UCSF campus, and returns with the drive from the previous month. Because TC and LR work in tandem, TC has most of the same files that LR has. He also backs up his work regularly. In this way, we essentially maintain a triple-plus backup system.

Document contents of incoming files

The documentation file PD_DOC.XLS has a tab SOURCE_MASTER_FILES. This identifies the file name on the password protected CD the agency sent, name of FHOP’s password-protected ZIP file, where documentation is stored, and any relevant notes or comments.

If the file arrives in SAS, a program (PD_DOC.SAS in D:\PDD\PGMS) calls macro CONTSRPT to do PROC CONTENTS of the incoming dataset(s), and store results in an Excel file on the Master Drive (E:\PDD\XLS\PD_DOC.XLS). This Excel file has one tab per master year or sets of incoming years. Table 1. shows results for 2009-2011 files. It highlights a few issues we address in standardizing the incoming file for longitudinal research.

Table 1. Example PD_DOC report

NAME	LABEL	_2009	_2010	_2011
ADMTDATE	ADMISSION DATE	D8 MMDDYY	D8 MMDDYY	D8 MMDDYY
ADMTDAY	DAY OF WEEK OF ADMISSION	c1		
ADMTMTH	MONTH OF ADMISSION	c2		
ADMTYR	YEAR OF ADMISSION	c4		
AGDYADM	AGE IN DAYS AT ADMISSION	n8	n8	n8
AGDYDSCH	AGE IN DAYS AT DISCHARGE	n8		
AGYRADM	AGE IN YEARS AT ADMISSION	n8	n8	n8
AGYRDSCH	AGE IN YEARS AT DISCHARGE	n8		
BTHDATE	DATE OF BIRTH (DOB)	D8 MMDDYY	D8 MMDDYY	D8 MMDDYY
DSCHDATE	DISCHARGE DATE	D8 MMDDYY	D8 MMDDYY	D8 MMDDYY
ETHNCTY	ETHNICITY	c1	c1	c1
ETH_RACE	CONCATENATED ETHNICITY/RACE GROUP	c2		
LOS	LENGTH OF STAY	n8		
LOS_ADJ	ADJUSTED LENGTH OF STAY	n8		
ODIAG1	OTHER DIAGNOSIS 1	c5	c5	c5
ODIAG10	OTHER DIAGNOSIS 10	c5	c5	c5
ODIAG11	OTHER DIAGNOSIS 11	c5	c5	c5
ODIAG12	OTHER DIAGNOSIS 12	c5	c5	c5
ODIAG13	OTHER DIAGNOSIS 13	c5	c5	c5
ODIAG14	OTHER DIAGNOSIS 14	c5	c5	c5
ODIAG15	OTHER DIAGNOSIS 15	c5	c5	c5
ODIAG16	OTHER DIAGNOSIS 16	c5	c5	c5
ODIAG17	OTHER DIAGNOSIS 17	c5	c5	c5
ODIAG18	OTHER DIAGNOSIS 18	c5	c5	c5
ODIAG19	OTHER DIAGNOSIS 19	c5	c5	c5
ODIAG2	OTHER DIAGNOSIS 2	c5	c5	c5
ODIAG20	OTHER DIAGNOSIS 20	c5	c5	c5
ODIAG21	OTHER DIAGNOSIS 21	c5	c5	c5
ODIAG22	OTHER DIAGNOSIS 22	c5	c5	c5
ODIAG23	OTHER DIAGNOSIS 23	c5	c5	c5
ODIAG24	OTHER DIAGNOSIS 24	c5	c5	c5
ODIAG3	OTHER DIAGNOSIS 3	c5	c5	c5
ODIAG4	OTHER DIAGNOSIS 4	c5	c5	c5
ODIAG5	OTHER DIAGNOSIS 5	c5	c5	c5
ODIAG6	OTHER DIAGNOSIS 6	c5	c5	c5
ODIAG7	OTHER DIAGNOSIS 7	c5	c5	c5
ODIAG8	OTHER DIAGNOSIS 8	c5	c5	c5
ODIAG9	OTHER DIAGNOSIS 9	c5	c5	c5

VARIABLE LENGTH. OSHPD’s numeric variables have the default length of 8 bytes, when 3-5 bytes is sufficient for most. Decimal variables are an important exception. They need to be length 8. Specifying a precise variable length saves space in the output file.

TIME-RELATED VARIABLES. Date of birth, admission and discharge are available, while other variables made from these are available only in 2009. We do not keep the constructed variables ADMTDAY, ADMTMONTH, ADMTYEAR. We can make them as needed since we have the relevant date variables. Note that OSHPD dates are numeric, while we prefer to receive date variables as character text strings, for reasons discussed elsewhere [3].

CLINICAL-RELATED VARIABLES. Finally, notice the order of other diagnoses (ODIAG). They go 1, 10-19, 2, 20-24, 3-9. When we standardize, we rename these variables DX01-DX24, assign the principal diagnosis to DX00, and assign the principal E- code the name ECD00. Then PROC CONTENTS will display in order. For ICD-10, we rename these variables DXT01-DXT24, assign the principal diagnosis to DXT00 and assign the principal E-code the name ECDT00.

Embedded in the controlling program, a tailored macro DOIT packs any additional E-Codes into the DX array then counts the number of DX and PX on the record. The resulting variables DXN and PXN also give some faint sense of illness severity. Renaming the array DX00-DX28 and storing the number with data are helpful when programming SAS arrays. It shortens processing time by only searching the array up to the max of DXN or PXN. Also, different years of incoming OSHPD files have different numbers of DX and PX variables, which macro DOIT addresses.

VARSXLS spreadsheet structure

Using documentation for a given year of data, we prepare a spreadsheet to standardize the incoming file longitudinally. The first row of the source VARSXLS spreadsheet (PDV.XLS) identifies column names, defined below. After preparing the spreadsheet, the macro VARSXLS calls this spreadsheet, which must have the .XLS extension. The SHEET = parameter specifies the sheet within the Excel file with the variable definitions used to process the data.

STORDER	The order variables will be stored in the output file. We group related variables. STORDER helps us to store variable groups as we want.
GROUP	This names the group where sets of variables belong. Grouping makes it easier to handle related variables consistently. It helps when reading PROC PRINTs.
VARNAME	This is the text name of a given variable. FHOP’s convention is to limit VARNAME length to 8 characters, to allow the final files to run in most computing environments. The macro permits text up to 32 characters long.
LABEL	FHOP requires mixed-case labels for all permanent variables to make it easier to produce report tables. An unlabeled variable may signal that a temporary variable was kept rather than deleted or that we have a typo.
LENGTH	If Type = N, length is between 3 and 8. Dates can be stored in a variable with length 4. Numeric decimal variables require length 8.
TYPE	C(haracter) or N(umeric)
FORMAT	FHOP requires mixed-case formats for categorical variables. Most formats have the same name as the variable. Some formats are for multiple variables. For these, we use a generic format name (HSA, HFPA, GEOG, $DX, $PX). Format labels begin with the value of the underlying string, for example, “1 Male”. This makes programming easier since the analyst does not have to remember the underlying value or its definition. It also maintains consistent order in listings. Absent a numeric prefix, frequencies will be alphabetical, which rarely is helpful.
OUT1	Y or blank. If Y this variable is kept on the main output data set.

OUT2	Y or blank. If Y this variable is kept on the second output data set.
OUT3	Y or blank. If Y this variable is kept on the third output data set.
OUTn	As many output files as needed can be specified.
SASCODE	Code to change the incoming data is entered here. Macros are permitted.
CODEORDER	Integer controlling order in which SASCODE is executed.
SORT	Blank or integer. If output is not a view, it will be sorted using a BY statement constructed from variables in order of their (non-missing) SORT parameter number.
STATS	This identifies basic descriptive statistics to be shown in the listing. Values for two SAS PROCS are permitted: F(requencies), U(nivariate). We constructed a special step to highlight (M)issing values. The LST file also includes PROC CONTENTS and a listing of 10 cases in the file.
YEAR COLUMNS	A required column whose name on Row 1 is an underscore followed by the 4-digit year parameter used in the RDYR macro call. Optionally, if the layout is the same over several years, one column can span multiple years, for example, "_2002_2006".

As the reader will see when viewing the various VARSXLS input files, datasets arrive with different names for the same variables. The VARSXLS file identifies incoming attributes, which may differ from year to year. These variations on a theme create the primary problem of the longitudinal researcher: consistency over time.

Contents of VARSXLS Year column

Variable name or column range in the input file. If reading in a text file using the INFILE = parameter, the Year column must contain specific ranges to identify variable locations on the input file record, for example the string ‘1250-1255’. If reading a SAS data set using the INSAS = parameter, the Year column must contain the name of a variable on the input data set, for example, from PD_DOC, the variable ODIAG1. The variable name in the Year column can be the same or different from the variable name in the VARNAME column on the same row. In the example, the VARNAME column will name the incoming variable DX01,

The word "CALC" (calculate). If CALC appears, the RDYR macro makes the variable that the column VARNAME specified instead of using a variable in the input file or data set. When CALC is used, SAS assigns values to a variable using SAS code or a macro in the SASCODE column. If the SASCODE column calls the macro DOIT, the controlling program must have SAS code for the macro DOIT, telling RDYR what to do.

Blank. In any year of an incoming file, a variable may or may not be present. When the Year column is blank, RDYR deletes the row and the variable on the row that was present in other years is not in the final file.

Document where the work is done

In Volume 1 of this series “The Basic Computing Environment”, we discussed a few programs central to our system [2]. They document project-specific information for a specific computer. Many research groups have multiple programmers working on the same project on different computers in different locations, and that describes FHOP. The authors work at geographically dispersed locations, to be specific, in different states.

It is important for programs to run easily across settings. The SETUP macro, copied into every major program, allows programs to run correctly with respect to inputs and outputs, regardless of how computers at different locations are organized. Because program testing and development occurs in both settings, we have found that this minimizes problems in coordinating work. It also sets up the possibility of an audit check, in that we can (and often do) compare results to verify they are the same.

```
*-----------------------------------------------------------------------------*
* Working environment                                                         *
*-----------------------------------------------------------------------------*;

%MACRO SETUP;
  %GLOBAL V R L;

%IF &ANALYST = TCLAY %THEN
%DO;
   %let V = E:\FHOP\PDD\XLS\PDV.XLS;             * Path to variable definitions;
   %LET R = I:\PDD\RAW;                          * Path to incoming data;
   %LET L = D:\FHOP\PDD\PGMS;                    * Path to PDyyyy.lst and .log;
   libname RAW "&R";
%END;

%ELSE %IF &ANALYST = LREMY %THEN
%DO;
   %let V = E:\PDD\XLS\PDV.XLS;                  * Path to variable definitions;
   %LET R = I:\PDD\RAW;                          * Path to incoming data;
   %LET L = D:\PDD\PGMS;                         * Path to PDyyyy.lst and .log;
   libname RAW "&R";
%END;

%MEND SETUP;

%SETUP;
```

THE RDYR MACRO

RDYR is a SAS macro. It uses an Excel spreadsheet (say PDV.XLS) assigned to the macro variable VARSXLS to control the process of importing files into FHOP's environment, applying standard variable names, lengths, and labels. It allows optional variable calculations.

All macros used must be available. AUTOEXEC.SAS in the study directory specifies their location. The preferred place is C:\TOOLS\FHOP (specific) or C:\TOOLS\GENLIB (generic).

A program (say PD.SAS) controls the sequence. After executing the SETUP macro and DOIT macro (if used), RDYR is invoked. The following describes the RDYR steps.

Invoke RDYR

In this part of the program PD.SAS, we prepare the RDYR macro to give control to the VARSXLS spreadsheet. RDYR imports the Excel file into SAS, and executes SAS code embedded in it to do transformations, sorts, and obtain basic descriptive and diagnostic statistics. RDYR needs to know if data are coming in as SAS or flat files, where the controlling spreadsheet and tab containing the needed information is located, and the year column in the VARSXLS sheet that will tell SAS about the incoming file contents.

The RDYR macro executes SAS code as specified. The code can be either regular SAS code typed into the spreadsheet, or it can call other macros to do tasks. We *code what we want to do the same – every time – in the VARSXLS input spreadsheet.* Called macros typically are located in standard macro libraries referenced earlier. However, the analyst also can write code in the macro DOIT embedded in the controlling program.

For OSHPD files, we tell SAS to output three files. DX&yyyy contains YR_OBS, DX(T)08-DX(T)&maxdx, and for PDD from 1995 forward, DXP08-DXP&maxdx (DX present at admission, yes, no). PX(T)&yyyy contains YR_OBS, PX(T)04-PX(T)&maxpx, and PXDT04-PXDT&maxpx (procedure dates). The MAIN file contains YR_OBS and all other variables. We use YR_OBS to link datasets when we need to call the full array of diagnoses (DX) or procedures (PX).

```
%rdyr(insas = &IN1,infile = &IN2, varsxls = &V, sheet = Define, year = &yyyy,
   out1 = ESAS.MAIN&yyyy, out2 = ESAS.DX&yyyy, out3 = ESAS.PX&yyyy,
   where2 = DXN gt 7, where3 = PXN gt 3);
```

The rationale for outputting multiple files is the amount of space we save. The incoming file is a huge matrix. The 2014 ED data takes 5,018 GB of space with most DX and PX empty. The total size of the three files we make is 2.2 GB, an orders of magnitude smaller. This translates into using less memory and less disk cache when programs call these files. We primarily work in the

MAIN files, and call the DX and PX files only when needed. Thus, most programs execute more quickly because we use less overhead.

RDYR outputs a lot of diagnostic information into both the LOG and LST files. The analyst must review these carefully to identify any possible errors. We also do a detailed review of basic diagnostic statistics: PROC CONTENTS, PROC PRINT, PROC FREQ, PROC UNIVARIATE. This helps to identify any important problems that might be lurking in the file we just imported.

Call RDYR

The highest-level macro DOYEAR controls the process, one year at a time. This part of the program changes annually, by adding the year and name of the incoming master file. After we successfully create a master file with confidential elements encrypted, and verify contents after a full review of the the LOG and LST, we use the asterisk (*) to comment out the task. In the example below, the year 2007 has no asterisk. It was the last year of data we ran at the time. We also delete the unzipped original unencrypted master, so it is not accessible to an unauthorized person. Note the different formats and names of files sent to us.

```
*%DOYEAR(yyyy = 1997, in = RAW.PD1997.TXT, sasortxt = TXT);
*%DOYEAR(yyyy = 1998, in = RAW.FHOP98, sasortxt = SAS);
*%DOYEAR(yyyy = 1999, in = RAW.DATA99, sasortxt = SAS);
*%DOYEAR(yyyy = 2000, in = RAW.OLIVA00, sasortxt = SAS);
*%DOYEAR(yyyy = 2001, in = RAW.PDD2001, sasortxt = SAS);
*%DOYEAR(yyyy = 2002, in = RAW.DHSQUINN02R, sasortxt = SAS);
*%DOYEAR(yyyy = 2003, in = RAW.CON_PDD03, sasortxt = SAS);
*%DOYEAR(yyyy = 2004, in = RAW.DHSQUINN04, sasortxt = SAS);
*%DOYEAR(yyyy = 2005, in = RAW.DHSQUINN05, sasortxt = SAS);
*%DOYEAR(yyyy = 2006, in = RAW.DHSPDD06, sasortxt = SAS);
%DOYEAR(yyyy = 2007, in = RAW.CDPH_PDD07, sasortxt = SAS);
```

Understand RDYR syntax

```
%MACRO RDYR(INFILE = , INSAS = , VARSXLS = , SHEET = Define, VARSDATA = , OUTLIB = ,
   PREFIX = , YEAR = , VIEW = N , suffix = );
        INFILE   = File name to be read in
        INSAS    = SAS data set to be read in
        VARSXLS  = Name of Excel file with variable names and input cols
        SHEET    = Name of worksheet in the above where names and cols are located
        YEAR     = Year of data file to be read in
        OUTLIB   = Libname for the final stored files
        PREFIX   = 2-3 letter prefix for output files and .LST file
        VIEW     = Y/N. If Y, macro creates a view, does not sort or do statistics
        SUFFIX   = Text to go after the year on the output data set
```

Specify input and output files

Input Files

Input to RDYR can be a text file, an Excel file, or a SAS data set. To read in a text file, specify ‘INFILE = ‘. To read in a SAS data set, specify ‘INSAS = ‘.

Output Files

Main SAS output data set. RDYR constructs the name of the main file as follows: The OUTLIB parameter followed by a period, followed by the PREFIX parameter, then the YEAR parameter then the (optional) suffix parameter. If VIEW = Y, the output is a view, created inside the macro by appending ‘/ view = ‘ followed by the data set name. The main output data set contains variable marked Y in the MAIN column in the variables spreadsheet, explained below.

Confidential SAS output data set. If the program sends confidential data elements to another file, the name of that output data set is the same as the main data set, except that the letter “C” is inserted between the PREFIX and YEAR. For example, the file BCCyyyy contains encrypted confidential variables such as names and addresses. The confidential output data set contains variables marked Y in the CONFID column in the VARSXLS spreadsheet, explained below.

Listing and log file, named PREFIX followed by YEAR followed by “.LST” and “.LOG”.

Note about the SUFFIX parameter: The FHOP convention is to have a suffix of “_V” at the end of the name of a view file. This naming convention is not built into the macro. So, when specifying VIEW = Y we also specify SUFFIX = _V.

The value of YEAR, with an underscore prefix (_1994), must be a variable (column heading on row 1) in the VARSXLS spreadsheet.

How RDYR works

Step 1: SAS reads in the VARSXLS spreadsheet, finds the Year column corresponding to the value of the YEAR = macro parameter, and deletes blank rows in that column.

Step 2: SAS uses information in VARSXLS to generate a data step defining all desired variables in a series of ATTRIB statements. These specify the desired variable names, types, lengths, formats, and labels.

Step 3a: (Reading a text file). SAS generates an INPUT statement reading values from the specified column locations directly into the desired variables.

Step 3b: (Reading in a SAS data set). SAS generates a SET statement, with a RENAME = option renaming input variables to be VAR1, VAR2, VAR3, etc. This is followed by a series of statements assigning the first desired output variable from VAR1, the second from VAR2, etc. This trick allows the type and length of the input variable to be different from the output variable even if the name is the same.

Step 4: SAS executes contents of the SASCODE column in the order specified in column CODEORDER. In preparing contents of the SASCODE column, any macro name without a specified parameter is modified to have the name in the VARNAME column as a parameter.

Step 5: SAS generates output data sets (or views), each with the appropriate variable list. This actually is a tailoring of the DATA statement at the beginning of the data step, but we think of it as happening last. Any variables with Y under OUT1 are on the KEEP list for the main output, and variables with Y under OUT2 are on the KEEP list for the second output. A file with confidential variables is always the last.

Step 6. SAS carries out post-dataset creation steps like sorting. If we sort the dataset, SAS makes YR_OBS after sorting. Then it outputs the various datasets. Finally, SAS produces basic descriptive statistics (PROC CONTENTS, PRINT, FREQ, UNIVARIATE) as VARSXLS directs.

What RDYR does

It is important to understand what this macro does. Table 2 is an example of how the incoming VARSXLS file might look.

Table 2. Example VARSXLS spreadsheet

								CODE			
VARNAME	LABEL	LENGTH	TYPE	FORMAT	OUT1	OUT2	SASCODE	ORDER	SORT	STATS	_1994
ssn	Social Security Number	9	C								21-29
SSNC	Social Security Number (Encr)	9	C		Y	Y	%MAKESSNC(SSN=SSN, DOBC=BTHDATEC);			M	CALC
RACE	Race/ethnicity	3	N	RACE.	Y					F	30
sexc	Sex m/f	1	C								31
SEX	Sex	3	N	SEX.	Y		%SEXNUM(sexc,sex)			F	CALC

In the column VARNAME, variables *ssn* and *sexc* are in low case and the others are upper case. This signifies that *ssn* and *sexc* are temporary, and blanks in columns OUT1 and OUT2 confirm this. We import these variables and transform them, specifically, converting *ssn* to SSNC and *sexc* to SEX. The RACE label indicates this variable combines race and Hispanic ethnicity. The final main output file (OUT1) will contain SSNC, RACE, and SEX. The confidential file (OUT2) will contain only SSNC.

If the macro call had parameters Year = 1994, View = N, Prefix = PD, outlib = SAS, then RDYR would generate the following data step:

```
Data
   SAS.PD1994 (keep = SSNC RACE SEX)
   SAS.PDC1994 (keep = SSNC);

   Attrib ssn  length = $9 label = ”Social Security Number”;
   Attrib SSNC length = $9 label = ”Social Security Number (Encr)”;
   Attrib RACE length = 3  format = race. label=”Race/ethnicity”;
   Attrib sexc length = $1 label “Sex m/f”;
   Attrib SEX  length = 3  format = sex. Label = ”Gender”;
   Input  ssn 21-29 RACE 30 sexc 31;
   %MAKESSNC(SSN = ssn, DOBC = BTHDATEC);
   %SEXNUM(SEX, sexc)
run;
```

To encrypt the 9-character social security number, macro MAKESSNC calls the encryption macro ENCSTR, which in turn calls a confidential key defined outside the macro. We do not make this encryption key public. Other users of this macro have to develop their own key, and **test, test, test.** Be absolutely certain that what goes in is the same as what comes out. Otherwise, **MAJOR** problems. We discuss elsewhere the limitations of OSHPD’s linkage variable RLN and the Medical Record number when it is available. [3].

We use ENCSTR to encrypt other confidential variables such as names and addresses, while MAKESSNC includes SSN-specific checks that need BTHDATEC and other variables, not shown in the example.

The SEXNUM macro makes a numeric sex variable, the first parameter is from a character sex variable, and the second parameter is the numeric code to assign. Here is the macro text.

```
%sexnum (from, to)
   if &from = ’M’ then &to = 1;
   else if &from = ’F’ then &to=2;
%mend;
```

In the example, the order in which MAKESSNC or SEXNUM executes does not matter, so we do not have to give values to the CODEORDER column in the variables spreadsheet.

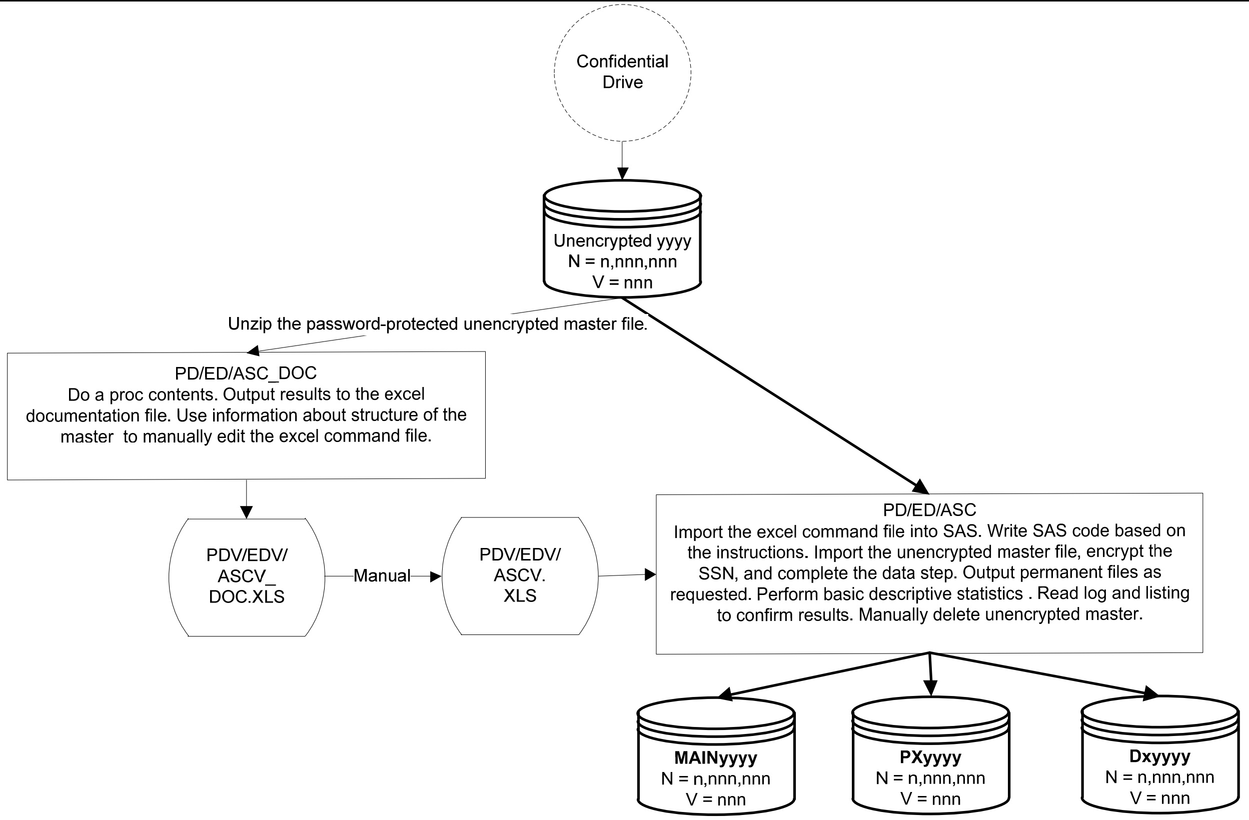
Note that the ATTRIB statement creates *sexc*, the INPUT statement gives a value to *sexc*, SEXNUM uses *sexc*, and *sexc* is not retained in either output dataset. SEXNUM assigns the variable SEX a numeric value. This variable does not exist when the data come into SAS, but SAS makes it because CALC appears in the _1994 column. If the cell now containing CALC was blank, the macro would not calculate the row for SEX.

Overview of Steps to Create Master Files

Figure 1 is from a Visio diagram showing steps to convert OSHPD's incoming confidential master files (PD, ED, ASC) into longitudinally consistent sets of files structured per FHOP standards. Unencrypted person-level data are stored on the confidential drive, in password-protected ZIP files. These are unzipped and documented in an Excel file (*_DOC.XLS), described earlier.

When data arrive as SAS files, we use information in the documentation file to edit the VARSXLS file driving the macro that reshapes the data into FHOP's structure. If data arrives as a flat text file, we use source documentation to edit the VARSXLS file.

Figure 1. Steps to create master files


Confidential Drive
Unencrypted yyyy
N = n,nnn,nnn
V = nnn
Unzip the password-protected unencrypted master file.
PD/ED/ASC_DOC
Do a proc contents. Output results to the excel documentation file. Use information about structure of the master to manually edit the excel command file.
PDV/EDV/ ASCV_ DOC.XLS
Manual
PDV/EDV/ ASCV. XLS
PD/ED/ASC
Import the excel command file into SAS. Write SAS code based on the instructions. Import the unencrypted master file, encrypt the SSN, and complete the data step. Output permanent files as requested. Perform basic descriptive statistics . Read log and listing to confirm results. Manually delete unencrypted master.
MAINyyyy
N = n,nnn,nnn
V = nnn
PXyyyy
N = n,nnn,nnn
V = nnn
Dxyyyy
N = n,nnn,nnn
V = nnn

CHECK LONGITUDINAL CONSISTENCY

Review contents report

After making all the files in a given set, we run CONTSRPT.SAS to verify that the files are internally consistent longitudinally. This program does a PROC CONTENTS over all available years, and outputs a temporary file with the variable names, labels, type (character or numeric), length, and format for each variable. The program merges these by year, then transposes, and merges again with the VARSXLS file that made the masters. This last merge retrieves the variables STORDER, GROUP, and VARNAME to display the variables in groups as originally defined. The final step outputs the information to an excel file for review.

Here our focus is if the same variable has the same label, type, length, and format in every year. Over time, some variables appear or disappear, and we check for such discontinuities. Early experience with these reports led us to develop the system we now use, so data can be consistent longitudinally. This report looked pretty sad when we began.

Table 3 is a cross-section of the PDD CONTSRPT.XLS for the period 1993-2000. Bear in mind that we have these files from 1983 forward. Variables are in the order STORDER established, from the source VARSXLS file, in this case PDV.XLS. Each yearly column shows the type: ((N)umeric, (C)haracter, (D)ate) and format associated with a given variable. Date variables are numeric length 4. Labels originally made in mixed case now are upper case. This is because CONTSRPT checks for multiple labels for the same variable, which upper case allows us to implement. Now that we think about it, we probably will update CONTSRPT to return to mixed case when it finds no problems.

Table 3. Example contents report

GROUP	NAME	LABEL	_1993	_1994	_1995	_1996	_1997	_1998	_1999	_2000
DEMOG	SEX	SEX	N3 SEX	N3 SEX	N3 SEX	N3 SEX	N3 SEX	N3 SEX	N3 SEX	N3 SEX
DEMOG	BTHDATE	DATE OF BIRTH	D4 DATE	D4 DATE	D4 DATE	D4 DATE	D4 DATE	D4 DATE	D4 DATE	D4 DATE
DEMOG	AGEADM	AGE AT ADMISSION (YEARS)	N3 AGE5CAT	N3 AGE5CAT	N3 AGE5CAT	N3 AGE5CAT	N3 AGE5CAT	N3 AGE5CAT	N3 AGE5CAT	N3 AGE5CAT
DEMOG	AGEADMD	AGE AT ADMISSION IF UNDER 3 (DAYS)	N3 AGEDAYS	N3 AGEDAYS	N3 AGEDAYS	N3 AGEDAYS	N3 AGEDAYS	N3 AGEDAYS	N3 AGEDAYS	N3 AGEDAYS
DEMOG	AGEDIS	AGE AT DISCHARGE (YEARS)	N3 AGE5CAT	N3 AGE5CAT	N3 AGE5CAT	N3 AGE5CAT	N3 AGE5CAT	N3 AGE5CAT	N3 AGE5CAT	N3 AGE5CAT
DEMOG	AGEDISD	AGE AT DISCHARGE IF UNDER 3 (DAYS)	N3 AGEDAYS	N3 AGEDAYS	N3 AGEDAYS	N3 AGEDAYS	N3 AGEDAYS	N3 AGEDAYS	N3 AGEDAYS	N3 AGEDAYS
RACETH	RACE	RACE/ETHNICITY	N3 RACE	N3 RACE						
RACETH	RACEN	RACE (1995)			N3 RACEN	N3 RACEN	N3 RACEN	N3 RACEN	N3 RACEN	N3 RACEN
RACETH	HISPANIC	HISPANIC ETHNICITY			N3 HISPANIC	N3 HISPANIC	N3 HISPANIC	N3 HISPANIC	N3 HISPANIC	N3 HISPANIC
TIME	ADMDATE	ADMISSION DATE	D4 DATE	D4 DATE	D4 DATE	D4 DATE	D4 DATE	D4 DATE	D4 DATE	D4 DATE
TIME	DISDATE	DISCHARGE DATE	D4 DATE	D4 DATE	D4 DATE	D4 DATE	D4 DATE	D4 DATE	D4 DATE	D4 DATE
TIME	YEAR	YEAR	N3	N3	N3	N3	N3	N3	N3	N3
TIME	LOS	LENGTH OF STAY IN DAYS	N3 LOSF	N3 LOSF	N3 LOSF	N3 LOSF	N3 LOSF	N3 LOSF	N3 LOSF	N3 LOSF
INOUT	SRCROUTE	SOURCE ROUTE	N3 SRCROUTE	N3 SRCROUTE	N3 SRCROUTE	N3 SRCROUTE	N3 SRCROUTE	N3 SRCROUTE	N3 SRCROUTE	N3 SRCROUTE
INOUT	SOURCE	SOURCE OF ADMISSION	N3 SOURCE	N3 SOURCE						
INOUT	SOURCEN	ADMISSION SOURCE (1995)			N3 SOURCEN	N3 SOURCEN	N3 SOURCEN	N3 SOURCEN	N3 SOURCEN	N3 SOURCEN
INOUT	SRCLICNS	SOURCE LICENSED UNDER			N3 SRCLICNS	N3 SRCLICNS	N3 SRCLICNS	N3 SRCLICNS	N3 SRCLICNS	N3 SRCLICNS
PAY	PAYSRC	PAYER SOURCE	N3 PAYSRC	N3 PAYSRC						
PAY	PAYSRCN	PAYER SOURCE (1995)			N3 PAYSRCN	N3 PAYSRCN	N3 PAYSRCN	N3 PAYSRCN		
PAY	PAYCAT	PAYER CATEGORY							N3 PAYCAT	N3 PAYCAT
PAY	PAYTYPE	PAYER TYPE							N3 PAYTYPE	N3 PAYTYPE
PAY	PAYPLAN	PAYER PLAN							C4 $PAYPLAN	C4 $PAYPLAN
PAY	TOTCHARG	TOTAL CHARGES	N5	N5	N5	N5	N5	N5	N5	N5

Variables SEX, BTHDATE, and age-related variables are in the group DEMOG. Notice that RACE discontinued in 1994, and a new race variable (RACEN) and separate variable for Hispanic ethnicity (HISPANIC) appeared in 1995. The INOUT group changed in the same year, while the PAY group changed in 1995 and 1999. Note that labels identify when what is essentially the same variable changes in some way. In the PAY group, TOTCHARGE is numeric length 5. Other numeric variables are length 3.

Summarize major longitudinal classifications

OSHPD files have several sets of clinical variables: principal and up to 24 DX, principal and up to 4 external E-Codes, and principal and up to 20 PX. From 1983 through 2014, OSHPD classified diagnoses using the International Classification of Diseases, 9th Revision, Clinical Modification (ICD-9), originally developed by the World Health Organization. The PDD also included the Major Diagnosis Category (MDC) and Diagnosis Related Group (DRG) based on the ICD-9 and after 2008, the Medicare Severity Diagnosis Related Groups (MS-DRG). The PDD uses the ICD-9 to classify procedures, while the ED and ASC use the Current Procedural Terminology (CPT), developed by the American Medical Association. Beginning with the 2015 data, PD, ED, and ASC diagnoses and PD procedures are classified using ICD-9 through September then ICD-10 forward. California death files have used ICD-10 since 1989. This change will require major revisions of many programs, formats and macros.

Sets of clinical codes change annually. As medical knowledge advances, some codes discontinue while others begin. At this point, having just created the master files, we are concerned primarily with identifying new codes, for which we need formats.

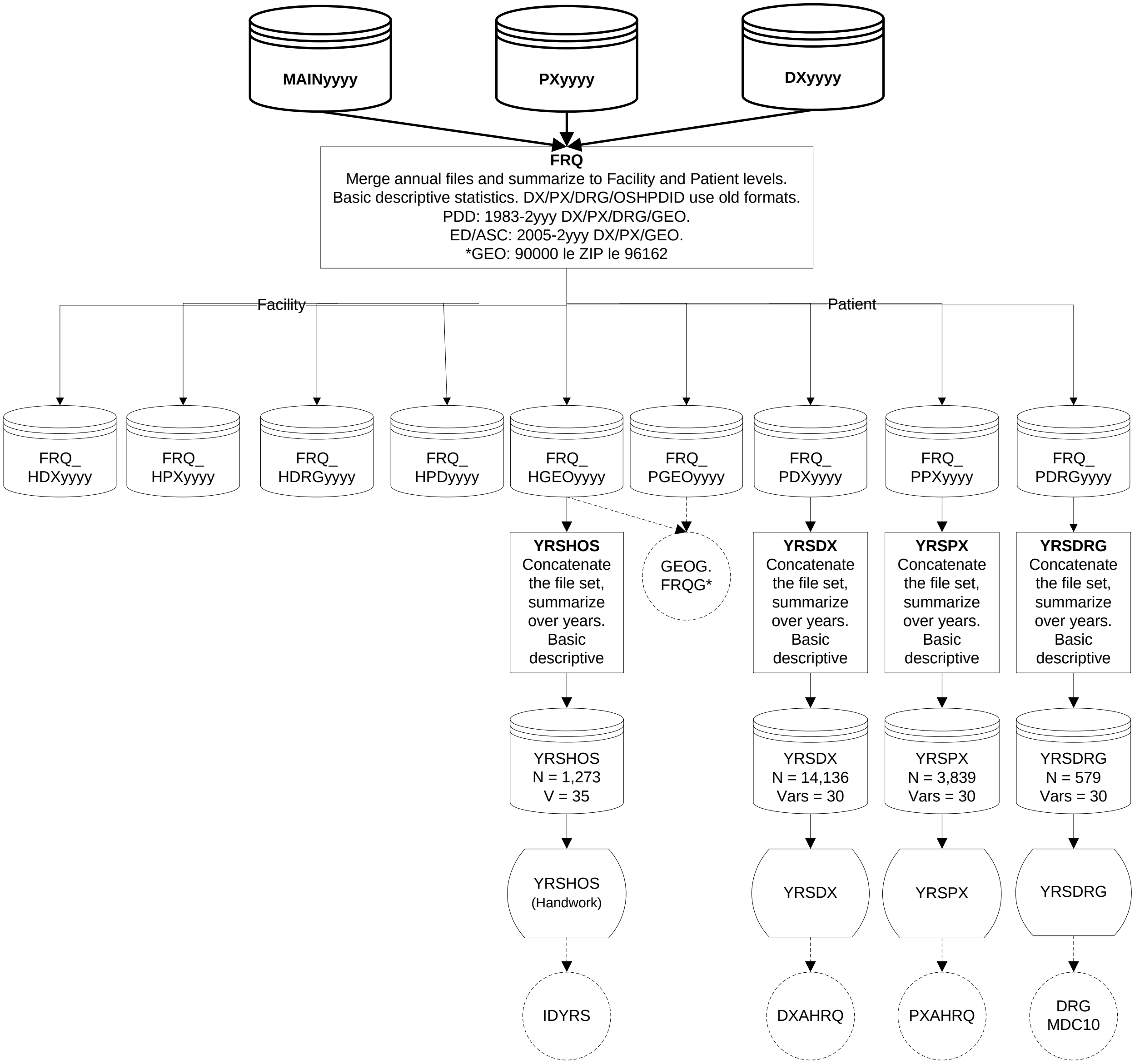
Hospitals also open, close, and change names. They may disappear when other hospitals in their area show huge increases in patients. This may reflect that the first hospital indeed closed, that it moved and OSHPD gave it a new identifier, or that both hospitals have the same owner, and OSHPD now allows them to submit consolidated reports [6]. At this point, ours is not to wonder why. We are interested only in identifying the appearance and disappearance of hospitals, and if we have a label for every identifier. Here we are not concerned with whether we have the *current* name, only *a* name.

Hospital structural capacity is a major area we begin to track as soon as we have master files. Hospitals filing PD records indicate whether patients were admitted through the ED, and the type of unit (general acute, psychiatric, rehabilitation, etc.) where they were treated. These are examples of licensable units with parallels in OSHPD's Hospital Annual Disclosure Report (HADR). Hospitals open and close these units over time, and these events are important to monitor for patient access to care [7].

Geographic variables are another important group to monitor longitudinally. ZIP-codes (ZIP) appear and disappear based on the needs of the United States Postal Service (USPS). New ZIPs appear. The USPS can divide a ZIP area, keep its number for a portion and assign a new number for the other portion, or disappear the entire ZIP and reintroduce it years later somewhere else. Where in the world is the ZIP? At this step in the process, we are interested only in identifying the universe of ZIPs and counties. We describe how we use these in our document on the geography master [8].

FRQ.SAS summarizes annual files from OSHPD and Vital Statistics to begin tracking these important changes in clinical, facility, and geographic characteristics. Figure 2 shows the data flow from master files through the creation of FRQ_* summary files, then the use of these files in the YRS*.SAS programs that make Excel files we evaluate for longitudinal consistency. Dashed circles indicate the next program calling these files, typically as the basis to begin making formats [9]. We use FRQ_ files with a named next program to begin the process of making formats. In the YRS* programs, we gather data summarized by FRQ.SAS to make a series of Excel files that enable us to review longitudinal changes such as the presence or absence of values and labels. We hand edit these and use as input to make formats.

Figure 2. Summary of steps to review longitudinal classifications


MAINyyyy
PXyyyy
DXyyyy
FRQ
Merge annual files and summarize to Facility and Patient levels.
Basic descriptive statistics. DX/PX/DRG/OSHPDID use old formats.
PDD: 1983-2yyy DX/PX/DRG/GEO.
ED/ASC: 2005-2yyy DX/PX/GEO.
*GEO: 90000 le ZIP le 96162
Facility
Patient
FRQ_ HDXyyyy
FRQ_ HPXyyyy
FRQ_ HDRGyyyy
FRQ_ HPDyyyy
FRQ_ HGEOyyyy
FRQ_ PGEOyyyy
FRQ_ PDXyyyy
FRQ_ PPXyyyy
FRQ_ PDRGyyyy
YRSHOS Concatenate the file set, summarize over years. Basic descriptive
GEOG. FRQG*
YRSDX Concatenate the file set, summarize over years. Basic descriptive
YRSPX Concatenate the file set, summarize over years. Basic descriptive
YRSDRG Concatenate the file set, summarize over years. Basic descriptive
YRSHOS N = 1,273 V = 35
YRSDX N = 14,136 Vars = 30
YRSPX N = 3,839 Vars = 30
YRSDRG N = 579 Vars = 30
YRSHOS (Handwork)
YRSDX
YRSPX
YRSDRG
IDYRS
DXAHRQ
PXAHRQ
DRG MDC10

Identify longitudinal discontinuities

FRQYRS.SAS outputs an excel file of the same name, with tabs for variable groups. Continuous variables such as age at admission are formatted. Here, we are looking for unformatted values, or sharp changes in distributions that might indicate variables read into SAS incorrectly. When we first ran FRQYRS for this document, we discovered we had not formatted age in some years, a humbling experience and more support for why we do this.

FRQYRS outputs both the number and percent of cases for each year. When definitions change (for example, RACE (1994) to RACEN (1995), we look for changes in the number of cases in a given category that might indicate definitional issues we need to address. Table 4 shows the number of cases for select variables in the DEMOG group, again focusing on the 1993-2000 cross section. It also highlights how variables measuring the same thing have different values and how the same numbers for the same concept have different meanings.

OSHPD separated race from ethnicity in 1995 and made two variables we named RACEN and HISPANIC. Compare original number of discharges with Hispanic ethnicity in RACE with HISPANIC. Notice that the number of people with Hispanic ethnicity continues its upward trend.

Table 4. Example FRQYRS report

VAR	LABEL	VALUE	DESCRIPTION	_1993	_1994	_1995	_1996	_1997	_1998	_1999	_2000
RACE	Race/Ethnicity	1	1 White	2,133,375	2,108,692						
		2	2 Black	322,168	319,210						
		3	3 Hispanic	911,444	893,942						
		4	4 AIAN	8,872	12,056						
		5	5 API	228,088	234,038						
		6	6 Other	39,543	37,739						
		7	7 Unknown	21,139	19,812						
RACEN	Race (1995)	1	1 White			2,639,782	2,673,719	2,725,193	2,761,659	2,792,395	2,797,482
		2	2 Black			323,291	319,268	321,383	322,507	327,259	328,963
		3	3 AIAN			15,765	15,890	16,053	17,369	14,964	13,852
		4	4 API			241,762	232,498	238,803	244,427	258,169	268,808
		5	5 Other			347,108	343,794	345,250	332,158	330,561	356,268
		6	6 Unknown			61,614	46,998	39,024	47,302	52,363	51,514
HISPANIC	Hispanic Ethnicity	1	1 Hispanic			911,950	932,762	945,625	946,330	985,101	1,010,829
		2	2 Non-Hispanic			2,552,720	2,600,138	2,658,407	2,698,775	2,709,361	2,714,197
		3	3 Unknown			164,652	99,267	81,674	80,317	81,249	91,861

Now turn attention to the variable RACEN. Rather than add a multi-race code, which Federal policies prefer and Vital Statistics enables, the OSHPD definition of “Other” became “other, multi-race (for multi-race patients not identifying a single preferred race), and natives of Central and South America”. The Federal policy groups natives of North, Central, and South America as “American Indian/Alaska Native” (AIAN).

In the combined race/ethnicity variable FHOP uses to calculate rates for its DataBook products, cases with race classified as Other or Unknown (which includes missing) rose overall from 1.6% in 1994 (with 1% the Federal standard for both race and ethnicity) to 3.7% in 1995 and continues upward thereafter, with wide variations by age. Vital Statistics follows Federal guidelines for classifying race and ethnicity. We discuss this issue in related documents [3,10].

Neither the California Department of Finance nor the Federal government publishes population denominators for the groups Other or Unknown. Realities such as these, with different types of problems across the various data sources, are why we agree with the Federal Government and recommend bridging race/ethnicity wherever possible [11-14]. *Calculating reliable longitudinal trends to monitor changes in California's race/ethnic disparities is extremely difficult because of inconsistencies within and across California datasets.*

Generate facility reports

The primary facility report is YRSHOS.SAS. Over the interval 1983-2017, 1,324 OSHPD facilities (PD, ED, ASC) reported 298,225,001 patient encounters. Reported facilities and patients increased substantially in 2005 when OSHPD introduced the ED and ASC. The equivalent birth certificate program report showed 549 facilities delivering 14,908,009 infants over the interval 1989-2016.

Generate clinical summaries

YRSDX.SAS gathers diagnosis summaries FRQ.SAS created from the OSHPD masters. We further summarize the data to get a total count over the period, and identify the first and last year the code appeared. The program outputs two sheets to YRSDX.XLS. One shows all diagnosis codes by year. The other identifies diagnosis codes without a label, which we must find. Over the interval 1983-2015, we found 15,143 unique ICD-9 codes for 892,995,223 diagnoses. After the conversion to ICD-10 on 01-Oct-2015, YRSDXT.SAS found 49,302 unique ICD-10 codes for 261,354,371 diagnoses over the interval 2015-2017.

YRSPX.SAS summarizes ICD procedure codes in the PD files and CPT procedure codes in the ED and ASC files, following the same process as YRSDX.SAS. Over the interval 1983-2015, we found 3,956 ICD-9 procedure codes for 201,472,225 procedures. After the 01-Oct-2015 ICD-10 conversion, YRSPXT.SAS found 82,236 codes for 14,019,498 procedures for 2015-2017.

YRSDRG.SAS summarizes DRG and MS-DRG codes in the PDD. Over the interval 1983-2007, we found 579 DRG codes for 93,266,457 patients. For the period 2008-2017, we found 764 MS-DRG for 38,932,121 patients.

Other documents on FHOP's website describe how we further use these files [7,9,15,16].

RESOURCES

Available VARSXLS Spreadsheets

For researchers interested in automating the production of master population health files standardized for longitudinal research, the following VARSXLS spreadsheets with associated SAS programs and sample files are available on request, with format libraries on our website [1]. The advantage of automating master file production is that other FHOP programs and macros will run with minimal modification. With RDYR and VARSXLS, it would take only a few weeks to make the transition.

File	From	Through	Name
Patient Discharge Data	1983	2017	PDV.XLSX
Emergency Department Data	1995	2017	EDV.XLSX
Ambulatory Surgical Center	1995	2017	ASCV.XLSX
Birth Statistical Master File	1989	2017	BCV.XLSX
Fetal Death Statistical Master File	1989	2017	FDTHV.XLSX
Death Statistical Master File	1980	2017	DTV.XLSX
Annual Hospital Disclosure Report	1980	2017	DATADIC.XLSX

Technical Support

All programs described here are available upon request. We strongly recommend that programmers join FHOP's SAS User group. FHOP has only two people who can provide a limited amount of handholding to learn how to use these programs. Users will have to contract for more than one hour of support.

ENDNOTES

1 See: http://fhop.ucsf.edu/data-management-methods.

2 Remy L, Clay T. (2018) Managing Longitudinal Research Studies: The Basic Computing Environment. San Francisco, CA: University of California, San Francisco, Family Health Outcomes Project. Available at: http://fhop.ucsf.edu/data-management-methods.

3 Remy L, Clay T. (2018) Managing Longitudinal Research Studies: Standardizing Variables Over Time. San Francisco, CA: University of California, San Francisco, Family Health Outcomes Project. Available at: http://fhop.ucsf.edu/data-management-methods.

4 National Institute of Standards and Technology (2001). Advanced Encryption Standard (AES). Department of Commerce, National Institute of Standards and Technology, Information Technology Laboratory (ITL). Last accessed 19-Nov-2018 at: http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.

5 See: https://technet.microsoft.com/en-us/library/dd835565(v=ws.10).aspx.

6 Remy L, Clay T. (2016) Managing Longitudinal Research Studies: Crosswalking Hospital Identifiers. San Francisco, CA: University of California, San Francisco, Family Health Outcomes Project. Available at: http://fhop.ucsf.edu/data-management-methods.

7 Remy L, Clay T, Oliva G (2004) Creating Hospital-level Datasets. San Francisco, CA: University of California, San Francisco, Family Health Outcomes Project. Available at: http://fhop.ucsf.edu/data-management-methods.

8 Remy L, Clay T. (2018) Managing Longitudinal Research Studies: Methods to Prepare the Geography Master. San Francisco, CA: University of California, San Francisco, Family Health Outcomes Project. Available at: http://fhop.ucsf.edu/data-management-methods.

9 Remy L, Clay T. (2018) Managing Longitudinal Research Studies: Maintaining OSHPD Format Library. San Francisco, CA: University of California, San Francisco, Family Health Outcomes Project. Available at: http://fhop.ucsf.edu/data-management-methods.

10 Remy LL, Clay T, Oliva G. (2011) Issues and Decisions to be made when Collecting, Coding and Reporting Race and Ethnicity for Public Health Indicators. Family Health Outcomes Project, University of California, San Francisco. Available at: http://fhop.ucsf.edu/data-management-methods.

11 Office of Management and Budget (2000). Provisional guidance on the implementation of the 1997 standards for federal data on race and ethnicity. Last accessed 19-Nov-2018 at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.175.7370

12 Office of Management and Budget (2000). Appendix C: The Bridge Report: Tabulation Options for Trend Analysis. Provisional guidance on the implementation of the 1997 standards for federal data on race and ethnicity. Last accessed 19-Nov-2018 at: https://obamawhitehouse.archives.gov/sites/default/files/omb/assets/omb/fedreg/r&e_guidance_notice.pdf.

13 Ingram DD, Parker JD, Schenker N, Weed JA, Hamilton B, Arias E, Madans JH. United States Census 2000 population with bridged race categories. National Center for Health Statistics. Vital Health Stat 2(135). 2003. Last accessed 19-Nov-2018 at: http://wonder.cdc.gov/wonder/help/populations/bridged-race/VitalHealthStatistics-Series2No135.pdf.

14 National Center for Health Statistics (2004). NCHS Procedures for Multiple-Race and Hispanic Origin Data: Collection, Coding, Editing, and Transmitting. Division of Vital Statistics, National Center for Health Statistics, Centers for Disease Control and Prevention. May 7, 2004. Last accessed 19-Nov-2018 at: http://www.cdc.gov/nchs/data/dvs/Multiple_race_docu_5-10-04.pdf.

15 Remy L, Clay T. (2018) Managing Longitudinal Research Studies: Birth and Fetal Death Statistical Master Files. San Francisco, CA: University of California, San Francisco, Family Health Outcomes Project. Available at: http://fhop.ucsf.edu/data-management-methods.

16 Remy L, Clay T. (2018) Managing Longitudinal Research Studies: Death Statistical Master File. San Francisco, CA: University of California, San Francisco, Family Health Outcomes Project. Available at: http://fhop.ucsf.edu/data-management-methods.