

## Extracting Clinical Research Data from OnCore® and Preparing it for Statistical Analysis in SAS®

Chang Xu, Dominic T. Moore, and Allison M. Deal,  
University of North Carolina Lineberger Comprehensive Cancer Center.

### ABSTRACT

This paper outlines a simple and straightforward method to extract data from the OnCore data base system and to prepare it for analysis using SAS. With this method, the user only needs to have basic online permission in OnCore along with basic knowledge of SAS and Excel. Example code and instructions are provided.

### INTRODUCTION

In 2010, we presented a method for extracting data from OnCore using SAS that required the user to have special permission to access the backend of OnCore, along with additional Oracle client software and special ODBC drivers<sup>1</sup>. This method works well for the few who fulfill these conditions. However, as our Cancer Center continued to grow, we found that we also needed a way to extract data from OnCore and create SAS datasets that was simpler and easier and therefore accessible to more users. The method discussed in this paper only requires basic online permission to access OnCore's BioStat console, and the knowledge and use of basic Excel, Base SAS fundamentals (i.e. PROC IMPORT and data step code) and PROC SQL. Additionally, we outline procedures for "best practices" to ensure that all data and programs are saved and cataloged appropriately.

The OnCore system (Online Collaborative Research Environment) is an Oracle® based data base and tool which provides an online environment for clinical research management, billing compliance, bio-specimen management, and patient registries management<sup>2</sup>. It is currently being used in many academic medical centers, cancer centers, and health care systems across the US. At UNC's Lineberger Comprehensive Cancer Center, we primarily use its clinical research management features to provide electronic case report forms (eCRF) and data collection/management for prospective clinical trials. With the growth of our Cancer Center and the amount of data being stored, we have seen a concomitant growth in the number and type of personnel who need to extract and analyze this data from OnCore. Not all of these analyses are for the final manuscript of a completed study. For example, demographic and adverse event data might need to be extracted for simple descriptive statistics used for monitoring purposes. The data extractors in these cases may be study coordinators and/or other personnel from the Clinical Protocol Office and not the study statistician of record. Thus, we have designed a simple and straightforward method that can be useful to many different users and at various stages during the study.

Some of the issues we run into when trying to get data from OnCore include:

- Data are in multiple eCRFs
- eCRFs have different structures (some stacked with multiple rows per patient (ex. prior therapies, drug administration forms) and some just one row per person (ex. demographics, baseline))
- Many extra variables get downloaded with each eCRF that are not needed
- Variable names are simply the text that was used as the question, and not ideal for use in SAS

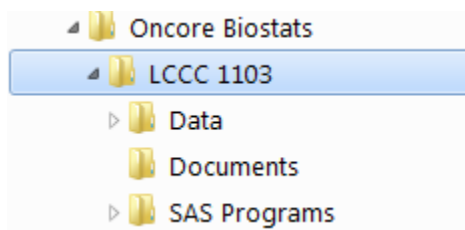
We wanted to automate as much of this process as we could, realizing that it would need input from the data management team along with the statistician/analyst who would be responsible for analysis. Our process does require one-time user input in deciding which variables are important to keep, and how to clean datasets to create variables that are needed for analysis. However, smart programming makes the task of updating the data easy – simply rerun the programs on the refreshed data in OnCore to create a final analysis dataset.

In summary, we have these steps:

0. Download files from OnCore and create Codebook.xlsx
1. Import raw xlsx data files into SAS
2. Output metadata to Codebook.xlsx and decide which variables to keep and rename
3. Import the new codebook for keep and rename
4. Create basic new variables (death dates, progression dates, summarize baseline history info into one row per patient)
5. Combine datasets for analysis

## INSTRUCTIONS

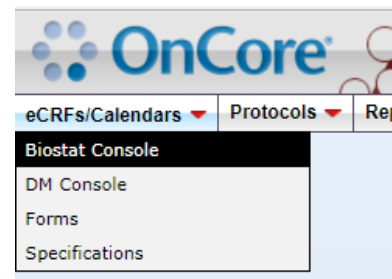
To start, create these folders in the study specific folder (Data, Documents, SAS Programs) (Display 1).



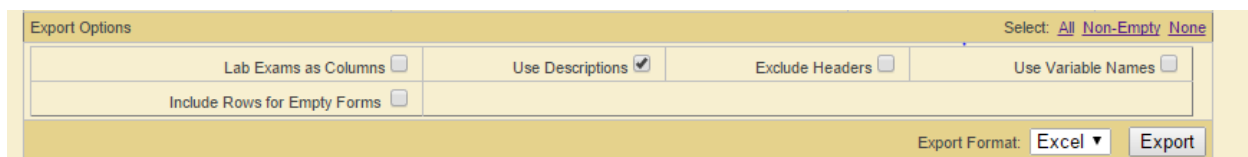
**Display 1 Recommended Folder Structure**

### STEP 0: DOWNLOAD FILES AND CREATE FIRST CODEBOOK

- a. Log into OnCore <https://oncore.unc.edu/>
- b. Go to “eCRFs/Calendars” and select “Biostat Console”.(Display 2)
- c. Enter the protocol number (ex. LCCC 1103) in “Select Protocol” blank.
- d. Select the protocol then go to “Data Export”, download all forms by “Select All” and check “Use Description”.(Display 3)

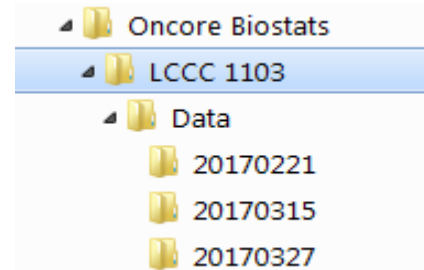


**Display 2 Biostat Console View in OnCore**



**Display 3 Data Export View in OnCore**

- e. Save all the excel files in the subfolder named by the downloaded date (ex. 20170327) in the “Data” folder under the protocol folder (ex. LCCC 1103). (Display 4)



**Display 4 Recommended Data Folder Structure**

- f. Create a blank excel workbook called Codebook.xlsx in your Data folder
- While you are still on the screen with “Data Export”, you will see a list of all the forms, and their name (Display 5). Copy the form names from this screen. Then, manually enter the name you want to use in SAS for each form. (This name should be SAS compliant)

|                                                    |                          |  |                            |  |                           |                          |
|----------------------------------------------------|--------------------------|--|----------------------------|--|---------------------------|--------------------------|
| <div>Select Protocol<br/>Type here to search</div> | Export Forms             |  |                            |  | Form Status: Multi-Select | Refresh                  |
| Accrual                                            | Form No.                 |  | Form Category              |  | Non-Empty Forms           | Export?                  |
| Forms »                                            | AdverseEvent(no DLT) V11 |  | Toxicity Assessment        |  | 40                        | <input type="checkbox"/> |
| SAE Counts                                         | BaselineForm V2          |  | Miscellaneous              |  | 60                        | <input type="checkbox"/> |
| Data Export                                        | Charlson V2              |  | Miscellaneous              |  | 32                        | <input type="checkbox"/> |
|                                                    | Chem-002 V7              |  | Complete Blood Chemistries |  | 549                       | <input type="checkbox"/> |
|                                                    | ConMeds V5               |  | Concomitant Medications    |  | 38                        | <input type="checkbox"/> |
| Protocol Calendar                                  | Demographics             |  | Demographics               |  | 67                        | <input type="checkbox"/> |

**Display 5 Forms Names View in OnCore**

- The Codebook.xlsx should then contain two columns: (Table 1)
  - “OnCore” for the form names directly as written in OnCore
  - “SAS” for their related dataset names.
- Save it with the sheet name “FORMNAMES”.

| OnCore                   | SAS      |
|--------------------------|----------|
| AdverseEvent(no DLT) V11 | AE       |
| BaselineForm V2          | Baseline |
| Charlson V2              | Charlson |
| Chem-002 V7              | Chem     |
| ConMeds V5               | ConMeds  |
| Demographics             | Demo     |
| Followup                 | Followup |
| SubsequentTherapy-001 V3 | SubT1_3  |
| SubsequentTherapy-001 V6 | SubT1_6  |
| SubsequentTherapy-001 V7 | SubT1_7  |

**Table 1 Example Form Names**

## STEP 1: IMPORT RAW XLSX DATA FILES

- a. Copy all SAS programs to the folder "Import" in the "SAS Programs" folder under the protocol folder.
  - Create dataset extract original
  - 1. Import oncore files
  - 2. Output codebook
  - 3-1. Import the new codebook for rename and keep
  - 3-2. Select variables to keep and rename
- b. Open the file "create dataset extract original.sas".
- c. Change the Protocol\_Num for different Protocols (ex. 1103).
- d. Change the date. "Date" is the downloaded date, which is the same as the folder name in STEP 0 (ex. 20170327).
- e. Run Step 1.

```
*****/
*****STEP 1. import oncore files*/
*****/;

%LET Protocol_Num =1103;      *indicate the protocol number;
%LET date          =20170327; *indicate the folder name;

%LET PATH = M:\Oncore Biostats\LCCC &Protocol_Num;
%put &path;
%LET sasd =&PATH\Data\&date;

LIBNAME A      "&PATH\Data";
LIBNAME sasd   "&sasd";

%include "&PATH\SAS Programs\import\1. import oncore files.sas";
```

- f. This code imports all forms listed in codebook.xlsx and creates \_raw\_ datasets with the names you want to use in SAS in the library SASD.
  - i. \_raw\_ae
  - ii. \_raw\_baseline
  - iii. \_raw\_charlson
  - iv. \_raw\_demo

## STEP 2: OUTPUT CODEBOOK AND DECIDE WHICH VARIABLES TO KEEP AND RENAME

- a. Run Step 2, to output codebook as sheet "FromSAS" into Codebook.xlsx.
- b. Create a new sheet "KeepRename" in codebook.xlsx by copying the sheet "FromSAS". (Table 2)
- c. Add two new columns "Keep" and "Rename" to indicate which variables you want to keep and how to rename them. This should be done manually with input from the statistician/analyst.
- d. Set Keep=1, if you want to keep this variable, otherwise no value. The value in the Rename column is the new name you want for this variable, otherwise no value. This name should be SAS compliant.

| These come straight from the SAS export |         |                                                                  |                                  |      | New columns added in Excel |                   |
|-----------------------------------------|---------|------------------------------------------------------------------|----------------------------------|------|----------------------------|-------------------|
| memname                                 | Var num | label                                                            | name                             | type | Keep                       | Rename            |
| _RAW_BASELINE                           | 1       | Sequence No#                                                     | Sequence_No_                     | char | 1                          |                   |
| _RAW_BASELINE                           | 2       | Initials                                                         | Initials                         | char |                            |                   |
| _RAW_BASELINE                           | 3       | Form                                                             | Form                             | char |                            |                   |
| _RAW_BASELINE                           | 4       | Form Desc#                                                       | Form_Desc_                       | char |                            |                   |
| _RAW_BASELINE                           | 5       | Not Applicable or Missing                                        | Not_Applicable_or_Missing        | char |                            |                   |
| _RAW_BASELINE                           | 6       | Phase                                                            | Phase                            | char |                            |                   |
| _RAW_BASELINE                           | 7       | Segment                                                          | Segment                          | char |                            |                   |
| _RAW_BASELINE                           | 8       | Cycle                                                            | Cycle                            | char |                            |                   |
| _RAW_BASELINE                           | 9       | Day                                                              | Day                              | num  |                            |                   |
| _RAW_BASELINE                           | 10      | Visit Date                                                       | Visit_Date                       | num  |                            |                   |
| _RAW_BASELINE                           | 11      | Form Status                                                      | Form_Status                      | char |                            |                   |
| _RAW_BASELINE                           | 12      | Arm                                                              | Arm                              | char |                            |                   |
| _RAW_BASELINE                           | 13      | Level                                                            | Level                            | char |                            |                   |
| _RAW_BASELINE                           | 14      | Histology                                                        | Histology                        | char | 1                          | hist              |
| _RAW_BASELINE                           | 15      | Date of diagnosis                                                | Date_of_diagnosis                | num  | 1                          | dx_dt             |
| _RAW_BASELINE                           | 16      | Stage of disease                                                 | Stage_of_disease                 | char | 1                          | stage             |
| _RAW_BASELINE                           | 17      | TNM Staging - T                                                  | TNM_Staging__T                   | char | 1                          | stage_t           |
| _RAW_BASELINE                           | 18      | TNM Staging - N                                                  | TNM_Staging__N                   | char | 1                          | stage_n           |
| _RAW_BASELINE                           | 19      | Performance status                                               | Performance_status               | char | 1                          | PS                |
| _RAW_BASELINE                           | 20      | Height                                                           | Height                           | num  | 1                          | ht                |
| _RAW_BASELINE                           | 21      | Height Units                                                     | Height_Units                     | char | 1                          | ht_units          |
| _RAW_BASELINE                           | 22      | HPV status                                                       | HPV_status                       | char | 1                          | HPV               |
| _RAW_BASELINE                           | 23      | Surgically resectable?                                           | Surgically_resectable_           | char | 1                          | surg_resect       |
| _RAW_BASELINE                           | 24      | Was the subject taking dexamethasone or other steroids within 30 | Was_the_subject_taking_dexametha | char | 1                          | within30_steroid  |
| _RAW_BASELINE                           | 25      | Was the subject taking any antihistamines within 30 days of trea | Was_the_subject_taking_any_antih | char | 1                          | within30_antihist |
| _RAW_BASELINE                           | 26      | Was the subject taking any H2 blockers within 30 days of treatme | Was_the_subject_taking_any_H2_bl | char | 1                          | within30_H2       |
| _RAW_BASELINE                           | 27      | Is the subject taking oral diabetic medications? (If yes, please | Is_the_subject_taking_oral_diabe | char | 1                          | oral_diabetic     |

**Table 2 Example of KeepRename tab in Codebook.xlsx**

### STEP 3: IMPORT THE NEW CODEBOOK FOR KEEP AND RENAME

- Run Step 3-1, to import the sheet "KeepRename" in codebook.xlsx.
- Run Step 3-2. \_clean\_ datasets will be created in library "SASD", only containing the variables you want to keep for each form and with the new names you want to use in SAS.
  - \_clean\_ae
  - \_clean\_baseline
  - \_clean\_charlson

- iv. `_clean_demo`
- c. One could stop here and have cleaned datasets to use as they see fit.

#### STEP 4: CREATE BASIC NEW VARIABLES

- a. Use program “Create dataset analysis.sas”
- b. First, automatically copy and sort all cleaned permanent datasets from SASD library to WORK library to manage.
- c. Examples are provided in the code, but the user should add as many as are needed – possibly one for each dataset.
- d. Once you have created any new variables you need, or collapsed stacked datasets into one row per patient, merge them all together to create a final cleaned dataset called `SASD._final_combine`

#### CONCLUSION

We have found that the techniques described above provide a variety of users a simple way to extract and manipulate data from an ongoing study for both monitoring purposes as well as final formal statistical analysis that takes place at the conclusion of a study. Additionally, variables can be given variable names that are more meaningful, and extraneous fields can be removed. With a small amount of time investment initially, users can easily rerun SAS programs to create datasets ready for analyses with the most up-to-date data.

Users can find all the code at the following website:

<http://unclineberger.org/research/core-facilities/biostats/allison-deal-ms>

#### REFERENCES

1. Chiu, Wing-Keung.(2010) Data Extraction from ONCORE® Using SAS. *SAS Global Forum 2010*. Available at: <http://support.sas.com/resources/papers/proceedings10/198-2010.pdf>
2. <https://forterresearch.com/enterprise-research-oncore/#faq>

#### CONTACT INFORMATION

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

Chang Xu

[Changxu@med.unc.edu](mailto:Changxu@med.unc.edu)

Dominic Moore

[Dominic\\_T\\_Moore@unc.edu](mailto:Dominic_T_Moore@unc.edu)

Allison Deal

[Allison\\_Deal@med.unc.edu](mailto:Allison_Deal@med.unc.edu)

## APPENDIX

### Create dataset extract original.sas

```
*****/
*****STEP 1. import oncore files*/
*****/;

%LET Protocol_Num =1103;      *indicate the protocol number;
%LET date          =20170403; *indicate the folder name that when the files
were downloaded.;

*Here we use the protocol_num specially for LCCC, users can change this as
they want it to be.;

%LET PATH = M:\Oncore Biostats\LCCC &Protocol_Num;
%put &path;
%LET sasd =&PATH\Data\&date;

LIBNAME A      "&PATH\Data";
LIBNAME sasd   "&sasd";

%include "&PATH\SAS Programs\import\1. import oncore files.sas";

*****/
*****STEP 2. output codebook.*/*
*****/;
* This step output the information of raw datasets as "codebook1.xlsx"
contains variable names, labels, types.;
%include "&PATH\SAS Programs\import\2. output codebook.sas";

/*****STEP 2-2. Copy all information in sheet FromSAS and manually create new
sheet "KeepRename" in "codebook.xlsx" in the data folder;
          Codebook.xlsx should contain new columns "keep" and "rename"
to indicate whether to keep and how to rename;*/

*****/
*****STEP 3-1. import the new codebook for rename and keep/
*****/;
* This step import the final "codebook.xlsx" to decide which variable to keep
and rename;
%include "&PATH\SAS Programs\import\3-1. import the new codebook for rename
and keep.sas";

*****/
*****STEP 3-2. select variables to keep and rename;*/
*****/;
* This step finalise the datasets in the library SASD with subject
"_clean_".;
%include "&PATH\SAS Programs\import\3-2. select variables to keep and
rename.sas";
```

## 1. import oncore files.sas

```
* Codebook contains oncore file name as "Oncore", and SAS dataset name as
"SAS".;
PROC IMPORT OUT= a.formnames
    DATAFILE= "&PATH\Data\Codebook.xlsx"
    DBMS=XLSX REPLACE;
    SHEET="FORMNAMES";
    GETNAMES=YES;

RUN;

DATA _null_;
    set sashelp.vtable;
    where UPCASE(libname)="A" and UPCASE(memname)="FORMNAMES";
    call symput('numforms', nobs);
RUN;
%put &numforms;
* macro variable numforms contains the numbers of total forms in oncore;

DATA formnames;
    set a.formnames;
    n+1;
RUN;
* use the datastep to do the iteration in macro;

options mprint symbolgen mlogic mfile;

%MACRO formnames();

%DO i=1 %TO &numforms;

data _null_;
    set formnames;
    if n= &i;
    call symput('Oncore', TRIM(Oncore));
    call symput('SAS', TRIM(SAS));
run;
%put &Oncore;
    PROC IMPORT OUT= sasd._raw_&sas
        DATAFILE= "&sasd\&oncore..xlsx"
        DBMS=XLSX REPLACE;
        SHEET="&Oncore";
        GETNAMES=YES;

    RUN;

%END;

%MEND;
%formnames();
* use the macro to import all files listed in the codebook.xlsx;
```

## 2. output codebook.sas

```
DATA vc;
    set sashelp.vcolumn;
    where UPCASE(libname)="SASD";
RUN;
*save the information of all variabls in library SASD.;

PROC SQL noprint;
    create table vc1 as
    select memname, varnum, label, name, type
    from vc;
QUIT;

PROC EXPORT DATA= vc1
    OUTFILE= "&PATH\Data\Codebook.xlsx"
    DBMS=xlsx Replace;
    SHEET="FromSAS";
RUN;
*select memname, varnum, label, name and type to output as "Codebook1.xlsx";
```

```
3-1. import the new codebook for rename and keep.sas
PROC IMPORT OUT= vc_keep
    DATAFILE= "&PATH\Data\Codebook.xlsx"
    DBMS=XLSX REPLACE;
    SHEET="KeepRename";
    GETNAMES=YES;

RUN;
```

### 3-2. select variables to keep and rename.sas

```
*keep;
DATA vc_keep2;
  set vc_keep;
  if keep*1 =1 ;
RUN;
* select only variable which is indicated to keep by "keep=1";

PROC SQL noprint;
  select SAS
  into : names
  separated by ' '
  from a.formnames;
QUIT;
%let datanames=%upcase(&names);
%put &datanames;

%MACRO keep();

  %DO i=1 %TO &numforms;

    %LET data_name = %SCAN(&datanames,%EVAL(&i));

    data vc_temp;
      set vc_keep2;
      where memname = "_RAW_&data_name";
    run;

    proc sql noprint;
      select count(*) into : nobs
      from vc_temp;
    quit;

    %IF &nobs > 0 %THEN %do;

      proc sql noprint;
        select name
        into : varlist
        separated by ' '
        from vc_temp;
      quit;

      %put &varlist;

      data &data_name;
        set sasd._raw_&data_name;
        id = Sequence_No_ +0;
        keep id &varlist;

      run;
    PROC SQL noprint;
      create table sasd._clean_&data_name as
        select id,*
        from &data_name(drop=Sequence_No_);
    QUIT;

  %END %DO;
```

```

        PROC SORT data=sasd._clean_&data_name;
        by id;
        RUN;

    %END;
%END;
%MEND;
%keep;

*rename;

DATA vc_keep3;
    set vc_keep2;
    where rename ne ' ';
RUN;
*select only variable which is indicated to be renamed.;

%MACRO rename();
    %DO i=1 %TO &numforms;

        %LET data_name = %SCAN(&datanames,%EVAL(&i));

        data vc_temp;
            set vc_keep3;
            where memname = "_RAW_&data_name";
            run;

        proc sql noprint;
            select count(*) into : nobs
            from vc_temp;
            quit;

        %IF &nobs > 0 %THEN %do;

        proc sql noprint;
            select name
            into : varlist_org
            separated by ' '
            from vc_temp;
            quit;

            %put &varlist_org;

        proc sql noprint;
            select rename
            into : varlist_rename
            separated by ' '
            from vc_temp;
            quit;

            %put &varlist_rename;

        %let word_cnt_re = %sysfunc(countw(&varlist_rename));
        %DO j=1 %TO &word_cnt_re;
            %let org_name = %qSCAN(&varlist_org,%EVAL(&j));
            %let new_name = %qSCAN(&varlist_rename,%EVAL(&j));

```

```

        data &data_name;
            set &data_name;
        rename &org_name = &new_name;
        run;

        %END;
        PROC SQL noprint;
        create table sasd._clean_&data_name as
            select id,*
                from &data_name(drop=Sequence_No_);
        QUIT;

        PROC SORT data=sasd._clean_&data_name;
        by id;
        RUN;

        %END;
    %END;
%rename;

```