

**From SAP to ADaM: The Nuts and Bolts**  
Nancy Brucken, i3 Statprobe, Ann Arbor, MI  
Paul Slagle, United Biosource Corp., Ann Arbor, MI

**ABSTRACT**

Now that you have data that is compliant with the SDTM standards, and the statistician has provided you with the analysis plan, how do you develop the analyses that will meet FDA guidelines and yet make developing tables simple? The answer, of course, is to create analysis datasets using ADaM specifications.

This workshop will introduce participants to the Analysis Data Model specifications version 2.1 and walk them through the creation of the ADaM datasets. Participants will create specifications and the programs needed to create the specifications. At the end of the workshop, they will create the metadata that is included with the submission.

**WHAT IS ADAM AND WHAT DOES IT MEAN FOR ME?**

**IN THE BEGINNING**

We are all familiar with the Study Data Tabulation Model (SDTM) created by the Clinical Data Information Standards Consortium (CDISC) back in 2004. This model, currently version 1.5 and described in the SDTM Implementation Guide (SDTMig) version 3.1.2, provides a stable model for capturing clinical data. By following this model, it becomes very simple to locate data for creating reports for a clinical trial.

According to the FDA, you are going to need both submission data and analysis data. Depending on the Statistical Analysis Plan (SAP), there are missing value imputations that will be needed, study population flags, and visit window calculations. There are also efficacy calculations, including both descriptive and inferential statistics, that need to be incorporated, with the ultimate goal being to get all of the data into a format that is "one proc away" between the analyzed data and the table generated.

How to make this happen has been a challenge for the CDISC community since the rollout of SDTM. There have been at least two analysis data models deployed by CDISC that has shown these struggles. These early models, released by CDISC, have shown a desire by the community to create "wide" models that provide variables for not only imputed data, but also for efficacy data. These initial versions of ADaM required multiple formats and continual updates and revisions by the standards committee. It also created problems for the adopting organizations, who, as you might suspect, always had one or two variables that wouldn't fit any of the existing variables in the model.

With the current release of ADaM version 2.1, the community adopted a primarily "long" version of the data. Rather than creating unique variables for each of the domains, a clinical trial was allowed to create multiple datasets containing a few variables, but many (many) rows. For a submission, each clinical trial is allowed to create as many datasets as is necessary to support the analytics being included.

In addition to the Basic Data Structure (BDS) datasets, there is one additional dataset required called the Subject Level dataset (ADSL). While this dataset does not follow the structure of the BDS, it is uniquely defined to contain all of the variables needed to describe each subject's involvement in a clinical trial.

**THE BASICS OF ADSL AND BDS**

The basic Subject Level dataset (ADSL) contains 10 variables for each subject in the dataset. These variables define the study, subject number, basic descriptors of the subject (like age and race), and treatment variables. To create this basic ADSL dataset, you will probably need at least the DM and EX domains from the SDTM model. However, to stop at these variables would be to shortchange the ADSL dataset. Other variables that you would want to use include the population flags and treatment start/end dates. Depending on your SAP, you will probably be pulling from many of the SDTM datasets in order to create these variables.

The remainder of the data can be incorporated into the Basic Data Structure. The three basic components of a BDS are the subject identifier, the data, and descriptors of that data. In its most basic form, each BDS can be created with only 7 variables-the study id, subject id (2), site id, treatment, parameter, and value.

By using just the parameter and value variables, you create the long structure of the BDS. The parameter variable is designed to contain the full description of the analysis. Examples of this would include "Average Weight" or "Final Systolic Blood Pressure". This describes what is being provided in either the AVAL (for numeric) or AVALC (for character) results. While these are the basics, the dataset should also include a couple of other types of variable that define the parameter. One of the variables should define whether the parameter is actual or derived while the other variable should define for derived variables what type of derivation occurred. Combined, these 4 variables can be used to capture any analysis used in a clinical trial.

Following this introduction, the remainder of this paper will provide more details on how these datasets are created.

## **WHAT ELSE IS NEEDED?**

Just like in SDTM, you are not only going to need the data itself, but metadata to describe what is being provided. This holds true with ADaM as well in that there are 3 metadata datasets required for each submission and one optional metadata dataset. The first two required datasets are similar to SDTM in that they describe the datasets and the variables in the datasets, much like the DEFINE.xml file.

The third required dataset works with the parameter field for each dataset. Since each unique parameter value in each dataset describes a result, this needs to be captured in a metadata dataset. Creating this isn't trivial, but it should provide enough details that when a reviewer looks at the data, it helps them understand how each parameter is created. The process involved is also described later in this paper.

## **WILL THERE BE MORE?**

While the basic ADaM structures involve the ADSL and multiple BDS datasets, there is some data that does not currently fit this format. Data such as adverse event results would be very difficult to analyze in a long format and make generating the tables "one proc away" virtually impossible. To this end, the CDISC community is working through creating custom domains for data such as adverse events, concomitant medications, and medical history.

## **THE UNDERLYING PURPOSE**

One of the goals of the CDISC community is to improve the clinical trial process. Providing data in using the ADaM data structures will facilitate that effort by making it easier for reviewers to find data used in the approval process, thus improving that process. Since the standard is being adopted globally, it will also make the results of clinical trials easier to submit to agencies outside of the US, but is there more.

By adopting ADaM to individual clinical trials, it is possible to make the integration of data across clinical trials easier. For ISS and ISE submissions, this can shorten the time from completion of the individual trials to final submission. In the long run, it may also facilitate pharmacovigilance monitoring of clinical trial data by making it easier to combine clinical trial data across indications.

## **I HAVE EVERYTHING, WHERE DO I START?**

### **THE SAP**

Alright. You've read through the protocol and SAP, taken a look at the annotated CRFs and TLF (table, listing, figure) shells, and are familiar with the ADaM implementation guide. Now what?

It's a good idea to read through the SAP again. The first time through should give you a high-level overview of what the study is about, the questions it is trying to answer, and the types of analyses being done. All of that information is crucial to understanding the ideas and purpose behind the TLFs. However, most SAPs are full of such details as how the study populations are defined, which visits will be analyzed, and which demographic parameters will be used as covariates in the primary and secondary efficacy analyses. Entire paragraphs, or even sections, of the SAP are devoted to these concepts. You may have picked up on some of this information the first time through, but it makes more sense the second time around.

The SAP is probably packed with other smaller, but no less important, definitions and derivations- for example, how to handle subjects who have multiple measurement values within a single visit window, or adverse events missing severity or relationship to study drug. These details are extremely important to consider when creating analysis datasets, and are very easy to overlook. One way to make them easy to locate in the future is to take a highlighter

with you during your second reading of the SAP (the highlight tool in Word is just as effective, if you prefer using the electronic versions of the study documents- just make sure you're using a copy of the SAP, and not the "official" version!), and highlight anything you find that relates to:

- Population definitions
- Visit windows
- Analysis covariates
- Handling of missing values (imputation rules, missing AE data, etc.)
- Handling of multiple records per visit
- Variable computations and derivations, including categorical variables
- Definition of treatment-emergent AEs
- Definition of baseline values
- Definition of prior/concurrent medications
- Handling of partial dates for AEs and medications

The list above is not meant to be comprehensive, but to give you an example of some of the information you will need when creating your analysis dataset specifications that can be found hiding in the depths of the SAP.

One other key piece of information is the study visit schedule (sometimes called the study event schedule). This is a chart showing exactly what information is collected at each protocol visit. It's nearly always contained in the protocol, and often copied over into the SAP. If it's not in the SAP (and even if it is), we highly recommend making a separate copy of it, and keeping it readily available.

## **TLF SHELLS**

The next step is to become familiar with the TLF shells for the study. These may be attached to the SAP, or in a separate file. As with the SAP, it helps to get a high-level understanding of what is going to be summarized or analyzed first, before diving into the details. Thus, it's good to take a look at the entire package first, before starting an in-depth, table by table, review of the document.

This second, detailed review is where you should start organizing which quantities will be needed for the various types of analyses you will be performing. Some properties that will help you sort through this information are:

- Subject-level vs. visit-level
- Appears on one or multiple types of tables
- Collected by itself or with other related variables

Using these properties to classify the quantities appearing on the TLF shells will help you create an initial grouping of your variables into datasets.

From your reading of the Analysis Dataset Model and ADaM Implementation Guide, you know that you will need, at a minimum, a subject-level analysis dataset, ADSL. All subject-level quantities, including demographic variables, analysis covariates, and population flags will be placed into ADSL.

You will also probably need one or more additional analysis datasets. There is no hard rule for determining how many analysis datasets you will need to create. You will need as many as are appropriate for your study- this is where the difficulty lies. How do you figure this out? Here's one method that has worked for us, at least.

Take out a piece of paper (either physical or virtual), and write down "ADSL", since you know already know that dataset is required.

Then, start at the beginning of the TLF shells (actually, anywhere in the package will work, but you may as well start somewhere), and go through them shell by shell. Record any subject-level variables that you see in the ADSL column on your paper. This list should include all variables summarized on demographic, baseline characteristics and study medication exposure tables, population flags, covariates listed in the models used on the efficacy tables, and phase/study completion status.

Adverse events and medications will also need to go into separate datasets. For AEs, take note of any demographic variables used in by-group summaries, and make sure those appear in your ADSL list. You will also most likely need a flag identifying treatment-emergent AEs, and possibly derived variables representing intensity, relationship to study drug, actions taken with study medication, and outcome, depending on how the AEs are summarized, and how these quantities are collected on the CRFs. If the tables can be generated using the categories collected on the CRFs,

then you probably will not need to create additional derived variables. However, if, for example, you need to summarize AEs by whether they are related or not related to study medication, and your CRFs collect relationship as definitely, probably, possibly, probably not or definitely not related to study drug, you should create a derived variable grouping the CRF values into a single related/not related flag variable.

Similar concepts also hold for medications. You may need to create flags indicating whether a medication is prior and/or concomitant, depending on how medications are being summarized.

The biggest part of your work lies in sorting out your efficacy data. Keeping in mind that the ADaM Basic Data Structure (BDS) dataset is structured as one record per subject, parameter and visit, start grouping the variables being summarized on the efficacy tables into different dataset headings on your paper. On your initial pass, keep variables summarized on your primary efficacy tables listed under the same dataset heading, and also include any other quantities collected with those variables which are summarized in a similar manner. For example, if your primary efficacy analysis uses a questionnaire score, which is one of several coming in from the same instrument, and all of them will be summarized by visit, all of them can be placed in the same analysis dataset. If other questionnaire scores were also collected, will be summarized by visit, and follow similar rules for imputation, they can either be placed in the same dataset, or held separately in another dataset.

ADaM does not mandate any particular correspondence between the number of unique tables/analyses and the number of derived datasets- a single dataset can support multiple tables/analyses. Thus, it is up to you to determine how to balance the ease of managing smaller analysis datasets with the time required for programming all of them.

Once you've made an initial pass through your TLF shells, you'll want to take another look at the groupings you have formed on the sheet of paper to see if any of them need to be modified. For example, if you have created several subject-level datasets, see if any of the variables they contain are actually covariates or demographic or grouping variables used on summary tables. If so, they should be moved into ADSL.

Examine your efficacy dataset groupings. Again, see if any of them should be combined due to similarities in collection, summarization, imputation or derivations. Alternatively, see if any of them should be split apart due to differences in the above attributes.

## **ADSL**

ADSL is the subject-level dataset required by ADaM. It contains population flags, treatment variables, and trial dates, as well as all of the covariates and demographic variables required by your TLFs. The ADaM Implementation Guide contains a complete list of the standard variables required in the dataset.

In addition to the basic study site and subject identifiers, ADSL must contain age, sex and race. It may also contain race groupings, if those are needed for any of your TLFs. There should also be a population flag defined for each study population shown on your TLFs. Compare the population flags noted on your list with the study populations defined in the SAP, and make sure they are all accounted for. At least 1 study population should be defined, and the study population flags should not be blank for any subjects in the dataset- all values should be either 'Y' or 'N'.

Which treatment arm variables should be included in your ADSL dataset depends on how your data will be summarized by treatment arm. For a single-period study, this is fairly straightforward- you will need a single variable representing the randomized (planned) treatment group, and possibly a variable representing the actual treatment group, if any subjects in your study received a different treatment than the one they were randomized to, and you need to summarize subjects by actual treatment on your safety tables. It's easier to include the actual treatment arm variable in your dataset at the beginning, and then remove it after the study is unblinded, if it turns out to be unnecessary, than to have to add it later.

For a crossover study, you will need to include a variable representing the planned treatment sequence, as well as variables representing the planned and actual treatment for each period of the study. If any of your TLFs summarize data by pooled treatment groups, or combinations of treatment groups ("all doses of Drug A", for example), then variables representing the pooled treatment groups should be included, as well.

The final set of required variables consists of trial dates- date of randomization, date of first exposure to study treatment, and date of last exposure to study treatment. If dose administration times are required for your TLFs, they should be included as well (in which case, you will probably also want to create datetime variables). If you have a crossover study, you should create variables representing the dates (and times or datetimes) of the first and last exposures to study treatment for each period. You may also need to create variables representing the starting and ending dates of each period, if those are different from the first and last exposure dates for the period.

Once you have accounted for all of the standard ADSL variables, add in the other subject-level variables shown on your list. You now have the complete set of variables required for the ADSL dataset in your study.

The next step is to work out the attributes and derivations for each variable. For this, you will need all of your study documents. The annotated CRFs and SDTM specifications will tell you which SDTM dataset contains the variables you will need. The SAP definitions that you have highlighted will tell you how to derive the additional variables.

Your DDT template should already contain the standard ADSL variables. Delete the ones that you do not need. Then, start filling in the variable labels, codelists/controlled terms and derivations as appropriate for your study. The demographic variables can probably be taken directly from the SDTM DM domain, though you may need to recomputed age as specified in the SAP. The various study populations should be defined in the SAP- you may need to work with the study statistician to determine exactly how to derive them for ADSL. For example, the Intent-to-Treat (ITT) population may be defined as all subjects who have received at least one dose of study medication and had at least one post-baseline visit. While it's probably fairly easy to determine if a subject received a dose of study medication, it may be trickier trying to decide if they had at least one post-baseline visit- you will need to decide which datasets you need to look through in order to find that post-baseline visit. Numeric treatment arm values make your programming easier if you number the treatment arms in the order in which they appear on your table shells. For example, assign a 1 to the first treatment arm, a 2 to the second treatment arm, and so on. Continue until you have filled in all of the attributes for all of your variables in the DDT. You may need to refine some of the definitions, as you encounter unexpected values in your data, but you should have a good start towards a complete ADSL at this point, along with the groundwork laid for your other analysis datasets.

## **WHAT ABOUT SOME REAL DATA, LIKE VITAL SIGNS?**

Now that you have reviewed the SAP and the table shells and created the ADSL dataset, it's time to start creating some "real" analysis data for the clinical trial. In your review of the SAP and table shells, you should have some idea of the calculations that will be needed for producing the reports. At this point, you should go back to that list of variables needed that you created earlier. Which variables are going to be needed to generate the analysis of vital signs data? Place those on your vital signs work sheet.

In addition to the analysis of vital signs data, you are going to need to identify which variables are going to be needed from ADSL to analyze the data. You will need some of the population flags that were created in ADSL, but probably not all of them. Identify which population variables from ADSL will be needed and add those variables to your vital signs work sheet. At this point, you also know you are going to need certain subject identification variables, so go ahead and add those to your vital signs work sheet.

You are also going to need treatment variables. One of the things you are going to notice is that, in the ADSL dataset, the variables are called TRTxxP but those same variables are not in the BDS definitions. Since the BDS is based on a specific point in time, you do not need to keep all of the treatment time points that you might have created in ADSL. Because of this, you are going to need to define how, based on the SAP, each result is going to be assigned to a treatment and assign this to TRTP (or TRTA if needed). Since this is a simple study, we will be assigning the treatment defined in TRT01P to TRTP.

You are also going to need to set some form of timing. How was the study conducted and what variables are going to be used for assigning the visit? Typically you are probably always going to be using the AVISIT variable, so we will add that to our spreadsheet. Often, vital signs are collected at multiple time points during a clinical trial. For this reason, you can use the ATPT variable to assign specific time points when the data is collected. There are also APHASE and APERIOD variables available for providing additional timepoint definitions.

Now, identify which variables from the SDTM VS domain you are going to need. These should be carried forward into the ADVS dataset as source variables. For each unique measurement value being carried forward, create a unique PARAM and PARAMCD value to be used in this dataset. Since everything in this analysis is numeric, you can assign their values to AVAL. By including these variables, you provide the reviewer the source data used for any calculation. There should be a one-to-one correspondence between the values of PARAM and PARAMCD. PARAMCD is also limited to 8 characters, to allow for easy transposition from the vertical structure back to a horizontal dataset required by some statistical procedures.

What about baseline and change from baseline values? In many of the "findings" type of tables, it is common to perform change from baseline calculations. For this reason, the BDS contains variables for capturing both the baseline values and change from baseline capture from the VS domain. You will need to identify, based on the rules outlined in the SAP, the rules for determining the baseline values. Once this is determined, add these variables and their method of calculating to your worksheet. For the record that is used for calculating the baseline value, the ABLFL value should be added with a value of 'Y'.

Sometimes subjects don't follow the protocol on their visits. Sometimes subjects show up more than once during their scheduled visit, creating an unscheduled visit. Because of this, rules are usually developed within the SAP for handling missing visits and for selecting visits within a specific time point. For cases when multiple visits occur during

a specific time point, an analysis flag should be added to the dataset designating which record is used for reporting. For this purpose, we will create the ANL01FL to designate records used for reporting. Note that the ADaM guidelines recommend keeping all of the records and visits found in your original SDTM dataset in the analysis dataset, as well, to make it easier for the reviewer to trace analysis records back to their source.

In some cases, the SAP will create a requirement for handling missing visit data. Often, the method used is Last Observation Carried Forward (LOCF). For this, we will be creating a value for AVAL that does not exist in the raw data from VS. Because of this, we will need to make some additional changes. First, the value of PARAMTYP will need to be set to 'DERIVED', since this value is calculated rather than coming from the raw dataset. You will also need to add a variable DTYPE to the dataset and, for cases when the value is derived, this should be set to the method for derivation. In this case, it should be set to 'LOCF'. Add these variables and definitions to your worksheet.

So, now that you have created both rows for the raw values and the imputed values, and created columns for change from baseline and analysis flags, you might be wondering how to determine when to add a row and when you should add a column? Within the ADaMig, there are 3 basic rules used for determining when a column is added and when a row is added. The implementation guide also provides quite a bit of detail surrounding the decisions. Rather than repeating that, let's boil it down to the basics. Go back to the fundamental BDS which consists of the subject, a parameter, a timing value, and a value. If you need to add an extra definition that describes the contents of the basic BDS, such as a flag or start/end date, then you add a column. If you are calculating anything from the AVAL or BASE columns, then you add a column. In all other cases, you create additional rows in the dataset.

It has been this guiding principle that has caused us to add the change from baseline variables instead of adding it as new row in the table. It is also for this reason when we calculated the derived values, we added them as rows rather than adding a new column.

This provides the basics for creating the ADVS domain used for our reporting purposes. For other domains, you may also need to additional derived variables or additional rows. Some instances of this include standardized values for labs, calculated scores for questionnaires, composite baseline values, and maximum/minimum values for scoring. For any special analysis that is used in reporting, the values used in generating those results should be in an analysis dataset. That is the purpose of "one proc away".

## **WHAT ABOUT NON-FINDINGS, LIKE AES?**

### **ADAE**

Congratulations! You've created the design for ADSL and for your efficacy findings datasets. What's next?

Looking at your paper and the TLF shells again, you probably see a stack of adverse event summary tables. And most likely, there are a number of tables summarizing treatment-emergent adverse events by System Organ Class (SOC) and Preferred Term, as well as by other attributes, such as severity and relationship to study drug; there may also be AE summaries by demographic variables, such as age group, race and sex.

Looking at your ADaM implementation guide, you find no mention of an ADAE dataset, other than a notation under Future Developments in section 1.3 that the ADaM team is working to create specifications for it. So, what do you do?

One approach would be to try to stuff the AE data into an ADaM BDS dataset. However, given the BDS structure of one record per subject, parameter and visit, that would force the AE data into a vertical structure, which is not at all what is needed for summarization. Since the ADaM philosophy is to create "analysis-ready" datasets, a vertical AE dataset just does not fit the model.

## **CONSTRUCTING THE UNDOCUMENTED SPECIFICATION**

The general practice for handling AE datasets has been to take the SDTM AE dataset, keeping the same structure of one record per subject and event, and then add any other variables required for generating your TLFs. You should be able to obtain a copy of the SDTM dataset specifications from whoever produced those datasets. Copy the AE specifications into your DDT. For traceability purposes (that is, to be able to trace the analysis dataset variables back to their original sources), it's recommended that any SDTM variables which are not modified in the analysis datasets should retain their original variable names and labels. Thus, simply copying the specifications for the SDTM variables into your DDT should be sufficient for handling them.

Then take a look at your variable list. If you are summarizing treatment-emergent AEs, you will need to add a treatment-emergent flag to ADAE; the definition for that should be provided in the SAP. It's entirely possible that

may be the only flag you will need to create. Occasionally, AEs are summarized by events related to study medication, and events not related to study medication. If your relationship to study drug variable contains only the values “related” and “not related”, no additional variable is needed. However, if your relationship to study drug variable contains values such as “Definitely”, “Probably”, “Possibly”, “Probably Not”, “Definitely Not”, and “Unknown”, you will need to create the Yes/No variable required for summarization.

You will also need to add in any ADSL variables required for producing your AE tables. At a minimum, you’ll need your treatment group identifiers. Date of first exposure to treatment will help support your treatment-emergent flag, so it should also be included. If this is a cross-over study, you should also pull over the variables containing date of first exposure for each period, as you will probably need to determine which treatment a subject was taking at the start of the AE.

Summaries of adverse events by demographic variables are less common in individual studies, but if your TLF shells contain such tables, you will also need to bring in those demographic variables from ADSL.

## **WHAT ABOUT THE METADATA?**

Now that you have created the ADaM datasets, you are not quite finished. Next comes the documentation, or metadata. Metadata is simply “data about your data”. As with the SDTM datasets, where you created the DEFINE.xml file to describe the contents, you will also be creating metadata with ADaM- but several times over. Then again, what was created in ADaM is much more complicated and requires more reference material. The first three sections are necessary in order to support the needs of the reviewer. The last section is optional and will only be discussed briefly.

### **THE ANALYSIS DATASET METADATA**

The analysis dataset metadata provides the first layer of the metadata. Each dataset that you create within the ADaM structure must be documented in this metadata file. For each analysis dataset, you will need to provide 6 additional pieces of information. This includes a description of the analysis dataset, its location, and the documentation for supporting the analysis in the dataset. The other 3 variables that are needed are required in all of the metadata datasets. These are: the structure of the dataset (for example, 1 record per subject, parameter and visit), the key variables in the analysis datasets (usually USUBJID, PARAMCD and AVISIT), and the class of the dataset. The dataset class describes the particular type of ADaM dataset. Currently, the only recognized classes are ADSL, BDS, or OTHER; future versions of ADaM will define additional dataset classes.

### **THE ANALYSIS VARIABLE METADATA**

Now that you have created metadata for each of the analysis datasets, you will need to create metadata for each of the variables in the analysis datasets. This metadata is probably the closest to the DEFINE.xml file created for the SDTM data. Since this is similar to other metadata you have created, we will simply refer to the examples in the implantation guide and the worksheet provided. Analysis value metadata includes such information as the dataset name, names, attributes, labels, types, display formats, codelists or controlled terms, and the source or derivation for each variable in all of your analysis datasets. The analysis dataset specifications that you customarily create for a study can serve as your analysis variable metadata- just make sure that you populate all of the required fields.

### **ANALYSIS PARAMETER VALUE-LEVEL METADATA**

Now comes the fun part of the ADaM Metadata. The analysis parameter value-level metadata serves to document information about the various parameters found in your BDS datasets. For each BDS dataset you have created, there should be a row in the Analysis Variable Metadata File for each value of PARAM/PARAMCD. This is where all the work we produced in the worksheets will now go. It will fit into the analysis variable metadata for each BDS dataset, with an additional Parameter Identifier column to capture parameter-level information describing the contents of the dataset.

How does this work? For each derived column or row that you have added to the BDS dataset, you will need to describe how it was created, and whether that derivation applies to all PARAMCD values in the dataset, or to most of them, or to a few specific PARAMCD values. For the variables other than the PARAM and PARAMCD entries, this information will probably be “\*ALL\*” is the descriptions in the other columns of the metadata file apply. For the PARAM and PARAMCD, the entries represent the different calculations that are used for the rows in the BDS. For

PARAM/PARAMCD entries that are the same for every row in the that specific BDS, you will probably assign the value of **"\*ALL\*"** to PARAM/PARAMCD as well. If you are using a predefined codelist, then you will probably want to use the **"DEFAULT"** value for the Parameter Identifier column. If you have unique derived entries in the PARAM/PARAMCD entries, then you will identify the specific name in the Parameter Identifier field and then define how this is derived in the remaining columns of the Analysis Variable Metadata. For example, a CHG variable, representing change from baseline, would be computed for all records, so it would be given a Parameter Identifier value of **"\*ALL\*"**, and its Source/Derivation would be given as AVAL – BASE. A derived total score for a questionnaire would have a Parameter Identifier value corresponding to the PARAMCD associated with that record (**'TOTSCORE'**, for example), and its Source/Derivation might indicate that it is computed as the sum of AVAL values for records with PARAMCD values of **'Q01'** through **'Q10'**.

In addition, you will also need to define all of the DTYPE values in your BDS dataset. If you are doing an LOCF imputation, you will need a row in your analysis variable metadata with the parameter(s) using the LOCF imputation defined in the Parameter Identifier column, **'DTYPE'** specified as the variable name (since this serves as the definition of the DTYPE field in the dataset), and **'LOCF'** specified as the Codelist / Controlled Term. The actual algorithm used for determining the LOCF value can be specified in the Source/Derivation column.

## **THE OPTIONAL ANALYSIS RESULTS METADATA**

This is not a required part of the ADaM submission, but probably comes the closest to capturing the analytics from the SAP, and is recommended as an aid to the reviewer. However, its contents should be determined jointly with the reviewer- it may contain information on all of the analyses performed for a study, or it may be limited to only the primary and secondary efficacy analyses. Basically, it serves as a link between your analysis datasets and the actual analyses carried out for the study. It should contain, for each analysis included, the table or figure number (could be the appendix number), the title of the table, the parameters included if a BDS dataset is used, the name of the analysis variable, the reason for performing the analysis (for example, **'Primary Efficacy Analysis as Pre-Specified in the Protocol'**), the dataset used, a description of any selection criteria or subsetting (for example, **'WHERE VISIT=3'**), a short textual description of the analysis (including the SAP section where it is defined, and a brief description of the statistical method employed), and the programming statements used to produce the analysis (the actual PROC statements).

For more details on this, we strongly recommend reviewing the ADaM specifications and discussing it's use and specifications with the agency reviewer.

## **CONCLUSION**

At this point, you now have enough background to create the basic analysis datasets for ADaM. That's the good news; the bad news is there is plenty more yet to do. As with any study, you will probably have datasets for exposure, concomitant meds, laboratory results, and additional findings datasets. In addition, you will probably discover that much of your work is based on your (and our) interpretation of what goes in each ADaM variable and its derivation. For some things, like treatment assignments, you will have many discussions with the others working on the study- this is all part of the process.

We have given you a good starting point, based on our experience, for the creation of basic ADaM datasets. As you progress, please be sure to reference the CDISC website for updates and their discussion board for others' interpretations.

## **REFERENCES**

ADaM version 2.1 and the ADaMig 1.0 can be downloaded from <http://www.cdisc.org>.

## **CONTACT INFORMATION**

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

Nancy Brucken  
i3 Statprobe  
5430 Data Court, Suite 200  
Ann Arbor, MI 48108  
734-757-9045  
Nancy.Brucken@i3statprobe.com

Paul Slagle  
United Biosource Corp  
2200 Commonwealth Dr, Suite 100  
Ann Arbor, MI 48105  
734-994-8940 x1604  
Paul.Slagle@unitedbiosource.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.