

Analysis Result Metadata... are we there yet?

Author: Carla Santillan

Chiltern International BVBA, Heverlee, Belgium

ABSTRACT

The creation of Analysis Result Metadata (ARM) will soon be a necessity for submission to the FDA. The use of macros to automate this might be the near future after the implementation of CDISC standard structures. Are we prepared to obtain ARM by pressing a button? We might be using CDISC Standard data structures, creating standard programs, almost auto-generate SDTM and ADAM data sets, and we use macros to reliably and efficiently create listings and figures. But in reality we are not done yet. Can we minimize extra work, re-use existing material, and standardize present tools towards the needs for the creation of Analysis Result Metadata? How can we benefit from this extra effort? And how do we gain speed and efficiency? We will explore these questions and more as we journey to analysis results metadata.

INTRODUCTION

In the near future we will need Define 2.0 in order to submit to the FDA following CDISC standards. Analysis Result Metadata is part of Define 2.0. A straightforward solution would be to take the metadata proposed by CDISC and enter it manually for each table or figure that is part of the analysis of primary and secondary endpoints of our project. This would then need to be repeated for the next project, and the next, and so on.

A more ambitious and probably more efficient approach would be to identify each element of this new metadata and its relationship with existing standards and to define new elements in the existing standards that will facilitate the creation of ARM and that might even automate its creation.

But how controlled are the elements involved in the creation of ARM, is it feasible to retrieve the needed information from each standard file? We will find some limitations and will show how we could cope with them.

STARTING WITH ANALYSIS RESULTS METADATA

The FDA announced on March 17, 2016 that the FDA support for Define.xml Version 1.0 will end for studies that start 12 months after March 15, 2017 (Ref. 1). Meaning that Studies starting after March 15, 2018 must use Define 2.0 (where study start date is informed consent date for first subject enrolled).

Once the decision of submitting the study following CDISC standards is taken, part of the ADaM metadata that will need to be prepared is ARM. Normally ARM is provided for the primary efficacy and secondary efficacy results. The content of this metadata is entered at the end of the process when the sponsor decides which tables or figures will be part of ARM. Even though it was already known which tables or figures it would be since the moment at which the TLF specifications were final.

A programmer in charge of any critical analysis has to identify the important, crucial, information needed - to program the analysis. This same information would be needed to generate ARM to assist the reviewer, therefore best practice would be that the critical analysis ARM is generated at the moment of programming.

At the moment creation of ARM is done at the latest stages, where we need to establish traceability between a given analysis result and the specific ADaM dataset and also provide information from the SAP and TLF specification with the respective links. As we experienced in each study there are many programmers working, and is not difficult to imagine that at the end of the process the original programmer of a critical analysis is not available (busy in another project, on holidays, moved to another company) therefore a new programmer should step in. For someone who was not involved in the programming of the critical tables included in ARM, this can be a challenge, even for the same programmer after some time it might be difficult to remember all the important details for each part of ARM.

The above implies some re work at the moment of ARM creation, not only for each of the analysis results part of this metadata, but also for the next study to be submitted and so on.

If for example an average of 15 to 20 analysis results are included per study in ARM. Let's see how much extra work we have with one analysis result. Note that some ARM fields can be 'easily identified' at first when looking at the analysis result, table/figure. Some other fields will involve re-read of Protocol or SAP, and review of TLF specifications for clarity. In the table below these last ones are marked as 'Entails re-work'.

The below table shows each ARM field's description and its source:

PhUSE 2016

Table 1: Analysis Result Metadata Field's Description and Source

Metadata Field	Description	Source
DISPLAY IDENTIFIER	An identifier for the specific analysis display (such as a display number), when combined with the result identifier provides a unique identifier for the analysis result.	Table/Figure name, in TLF metadata (*). Easy to identify
DISPLAY NAME	Title of display , including additional information that might be needed to describe and identify the display (e.g., analysis population).	Table/Figure name, in TLF metadata. Easy to identify
RESULT IDENTIFIER	Identifies specific analysis result when there are multiple analysis results on a display.	Table/Figure footnotes, in TLF metadata. Easy to identify
PARAM	The analysis parameter that is the focus of the analysis result.	From ADaM specifications. Easy to identify
PARAMCD	Corresponds to PARAM in the analysis dataset	From ADaM specifications. Easy to identify
ANALYSIS VARIABLE	The analysis variable being analyzed	From ADaM specifications. Easy to identify
REASON	The rationale for performing this analysis. It will indicate when the analysis was planned (e.g., 'Pre-specified in Protocol,' 'Pre-specified in SAP,' 'Data Driven,' 'Requested by FDA') and the purpose of the analysis within the body of evidence (e.g., 'Primary Efficacy', 'Key Secondary Efficacy', 'Safety').	Manual input from CP, from SAP or Protocol. Entails re-work, this information was clearly understood and available when the analysis was programmed
DATASET	The name of the dataset(s) used to generate the analysis result	From ADaM specifications. Easy to identify
SELECTION CRITERIA	Specific and sufficient selection criteria for analysis subset and / or numerator – a complete list of the variables and their values used to identify the records selected for the analysis.	Manual input from CP, from SAP or Protocol. Entails re-work, this information was clearly understood and available when the analysis was programmed
DOCUMENTATION	Textual description of the analysis performed.	Manual input from CP, from SAP or Protocol. Entails re-work, this information was clearly understood and available when the analysis was programmed
PROGRAMMING STATEMENTS	The software code used to perform the specific analysis. This will include the model statement (using the specific variable names) and all technical specifications needed for reproducing the analysis (e.g. covariance structure).	From SAS ® code, manual input from CP. Entails re-work, this information was clearly understood and available when the analysis was programmed

(*) TLF metadata: Describes basic parameters per analysis output (table, listing, figure) within a standardize template, like table number, title, footnote, etc.

Assuming that all our clinical programmer (CP) colleagues are familiar with the study and its ADaM metadata, particularly in Analysis Result Metadata, four of these eleven rows would need re reading parts of standard documents, and this would need to be done at least another 15 times for the rest of the ARM materials. It does not seem too difficult, but it is certainly time consuming and we would rather try to avoid it in the next study.

Especially since we know that all those field's contents were natural questions which were already answered by the Clinical Programmer at the moment of programming the relevant analysis result.

- Can we minimize these extra work?
- Can we re use existing material? And standardize present tools?
- How do we gain speed and efficiency?

MINIMIZE EXTRA WORK

The information needed for the Analysis Result Metadata does not come from one place only. This information comes from the SAP, TLF specifications, from ADaM metadata, and even from SAS programs. However, most of these source files are standard (CDISC or company standards), and within each standard we could identify the elements needed and specify the content in a way that is practical for the creation of ARM. This would minimize the work for the standard values, and we would mainly need to focus on new or unexpected values.

This section will show an example of Analysis Result Metadata and its relationship with elements in the existing standards. We presuppose the following standards: eCRF, SDTM and ADaM metadata, SAP template, standard TLF specifications, and a file containing titles and footnotes for the standard TLFs, referred to in this article as 'TLF metadata'.

The sponsor ensures consistency across studies and reduction of some work by using standards.

The use of a standard eCRF implies that most of the SDTM annotations can be available, some SDTM dataset template programs can be created.

In addition by having standard ADaM and TLF specifications, some standard macros and some TLFs template programs can be created, for which the code would be available to be adapted and re-used.

Standard Protocol and SAP template, ensure consistency in terminology and facilitate creation and understanding of the files.

The example below uses a case study to show the standard files involved in the creation of ARM and also the relationships between them that would be needed to automatically create ARM.

Example:

Among other things that are not relevant for this example, the study measured daily pain scores included in E-diary data. The data is mapped into SDTM.QS and then used in ADaM.ADQS. Where NRS, Numeric Rate Scale, questionnaire is used to give a numeric value to the intensity of pain (Ref.2).

As described in the protocol, one of the primary efficacy analysis variables is the percentage change from baseline to end of study (or early discontinuation) of the mean NRS pain score for overall pain. In the efficacy table chosen for this example, NRS scores were analysed using an ANCOVA model which included dose group and region (REG1 and REG2) as fixed factors and baseline NRS pain score of overall pain as a covariate. The analysis was conducted using the SAF population.

Figure 1: Analysis Display Layout for example

Program: </directory/directory/directory/program.sas>		XXXX Version		
Study: 99-NN-001		Table 12.3.1.1		
Mean NRS Pain Score Over the Last 5 Days for Overall Pain. Inferential Analysis		Source: XXXX		
		Full Analysis Set		
Study Part: xxxxxxxx				
Parameter	Statistic	Placebo (N=XXX)	DRUGA 3 mg (N=XXX)	DRUGA 5 mg (N=XXX)
EoT Change From Baseline	n	xxx	xxx	xxx
	Mean (SEM)	xx.xx (xx.xx)	xx.xx (xx.xx)	xx.xx (xx.xx)
	95% Two-sided CI	(xx.xx, xx.xx)	(xx.xx, xx.xx)	(xx.xx, xx.xx)
EoT Difference vs. Placebo	Mean (SEM) [2]		xx.xx (xx.xx)	xx.xx (xx.xx)
	95% Two-sided CI [2]		(xx.xx, xx.xx)	(xx.xx, xx.xx)
	p-value[1] p-value[2]	x.xxx	x.xxx	x.xxx
Linear Trend [3]	Estimate	x.xxxx		
	p-value	x.xxx		

SEM: Standard Error of the Mean. NE: Non-Estimable.
 From ANCOVA model that includes treatment group and region as fixed factors and baseline value as a covariate.
 [1] Overall treatment effect.
 [2] From the pairwise comparisons of each active group vs. placebo based on Dunnett's test in the ANCOVA model.
 [3] Overall treatment effect from linear trend.

Date ddMmmYYYY hh:mm:ss Page X of Y

PhUSE 2016

Based on the template of ARM (Ref. 3), information in ADaM metadata, Statistical Analysis Plan (SAP), Tables Listings and Figures (TLF) specifications, and TLF metadata the following ARM was created.

Table 2: Analysis Results Metadata, Example

Metadata Field	Metadata
DISPLAY IDENTIFIER	Table 12.3.1.1
DISPLAY NAME	Mean NRS Pain Score Over the Last 5 Days for Overall Pain. Full Analysis Set
RESULT IDENTIFIER	Treatment difference results (Mean, confidence interval, p-value)
PARAM	Overall Pain Score during the 5-day Period
PARAMCD	PLPNOV
ANALYSIS VARIABLE	CHG, BASE, TRT02AN, GEOREGN
REASON	Primary efficacy analysis as pre-specified in protocol
DATASET	ADQS
SELECTION CRITERIA	fas1fl='Y', paramcd='PLPNOV', trt01pn~=. , avisit='EoT'
DOCUMENTATION	See Protocol Section XX for details. Program: program_ex1.sas. NRS scores were analysed using an ANCOVA model which included dose group and region (REG1 and REG2) as fixed factors and baseline NRS pain score of overall pain as a covariate.
PROGRAMMING STATEMENTS	<pre> data pain; set adam.adqs; where fas1fl='Y' and paramcd="PLPNOV" and avisit="EoT"; run; proc mixed data=pain; class &trt georegn; model chg=base &trt georegn; lsmeans &trt/cl adjust=dunnett; estimate 'Linear trend' &trt -2 -1 0 1 2; ods output type3=pvalue; ods output lsmeans=lsm; ods output diffs=dif; ods output estimates=trend; run; </pre>

PhUSE 2016

The figure below shows each element of ARM and its source standard file. Considering that the following (CDISC/Sponsor) standard files are available: ADaM metadata, Protocol and SAP template, TLF specifications, TLF metadata.

Information can be found in each standard, ideally would link all these standard files programmatically. But some of our standard files, are Word® documents, which presents our first limitation. How can we minimise the difficulties this entails? Let's get a view of the type and content of each standard.

Figure 2: Analysis Result Metadata and Standard Files

Metadata Field	Metadata	
DISPLAY IDENTIFIER	Table 12.3.1.1	TFL Metadata
DISPLAY NAME	Mean NRS Pain Score Over the Last 5 Days for Overall Pain. Full Analysis Set	
RESULT IDENTIFIER	Treatment difference results (Mean, confidence interval, p-value)	TFL specification
PARAM	Overall Pain Score during the 5-day Period	ADaM Metadata
PARAMCD	PLPNOV	
ANALYSIS VARIABLE	CHG, BASE, TRT02AN, GEOREGN	Protocol/SAP
REASON	Primary efficacy analysis as pre-specified in protocol	
DATASET	ADQS	SAS code
SELECTION CRITERIA DOCUMENTATION	fas1fi='Y', paramcd='PLPNOV', trt01pn=~., avisit='EoT' See Protocol Section XX for details. Program: program_ex1.sas. NRS scores were analysed using an ANCOVA model which included dose group and region (REG1 and REG2) as fixed factors and baseline NRS pain score of overall pain as covariate.	
PROGRAMMING STATEMENTS	<pre> data pain; set adam.adqs; where fas1fi='Y' and paramcd='PLPNOV' and avisit='EoT'; run; proc mixed data=pain; class &trt georegn; model chg=base &trt georegn; lsmeans &trt/cl adjust=dunnett; estimate 'Linear trend' &trt -2 -1 0 1 2; ods output type3=pvalue; ods output lsmeans=lsm; ods output diffs=dif; ods output estimates=trend; run; </pre>	

Starting from TLF metadata, this is an Excel® file that contains (among other information) titles, footnotes and the source ADaM dataset used in each table, figure or listing. The type of output (dataset, table, listing or figure) and the name of the program producing it are also mentioned here.

See below a simplified version of our TLF metadata for this example:

Figure 3: TLF Metadata Simplified Version

Program name	Output Name	Output Type	Output Number	Title 3	Title 10	Footnote 2	Footnote 3	Footnote 9	Source	Program Location	Developer	Verifier	Comments
t_121121_sas	_2	Table	2.1	Analysis Sets in REG1		[1] All randomized subjects who took at least one dose of study medication (DRUGA or DRUGB or placebo).	[2] All randomized subjects who received at least one dose of double-blind study medication and who have a primary NRS pain score for overall pain	ADSL	directory/directory/program_name.sas		CPA	CPB	
example1	_1	Table	12.3.1.1	Mean NRS Pain Score Over the Last 5 Days for Overall Pain. Inferential Analysis		Includes all treatment groups and information until Visit 6 (included).	Last non-missing observation before first intake of study drug is used as baseline.	ADQS	directory/directory/directory/program_name.sas		CPA	CPB	
example1	_2	Table	12.3.1.2	Mean NRS Pain Score Over the Last 5 Days for Overall Pain. Inferential Analysis		Only includes subjects randomized to placebo in Part 1 and taking DRUGA in Part 2. Columns refer to DRUGA treatment from Visit 6 onwards.	Last non-missing observation before first dose of DRUGA is used as baseline.	ADQSAR	directory/directory/directory/program_name.sas		CPA	CPB	

PhUSE 2016

From this file two fields for ARM are available, the Output Number used as 'Display Identifier', and from 'title 3' to 'title 10' column used as 'Display Name' field in the metadata. There is also a relationship with the ADaM source dataset, see column 'Source', this is the field 'Analysis Variable' in ARM. The location of the program for this particular table is also present. Although the efficacy tables/figures are not identified in this file and we are missing, for example, the link between the table of interest and the 'result identifier' which is information found in the SAP.

Continuing with the well-known ADaM specifications, we can ensure for a BDS domain that the link between dataset, variable and PARAMCD is present.

Figure 4: ADaM Specifications

Variable ID	Domain of the dataset, used for grouping	UNFILTER Qualifying Variable	Variable Name	Variable Label	Variable Type (character)	Origin of the Variable	Order of	Variable Length	Variable Code List	Variable Comment
seq	domain	srcvar	varname	varlabel	vartyp	origin	cd	as	asformat	comment
388	ADQSAH	PARAMCD	BPISMEAN	BPI Pain Severeance Total Score	Num	Derived		8		PARCAT1='BPI SHORT FORM', PARCAT2='BPI PAIN SEVERITY', calculated per day from the pain severity parameters 'BPI203' to 'BPI206' as the average value of the 4 AVAL values. See SAP 6.2.2.4.
389	ADQSAH	PARAMCD	PLPNOV	Overall Pain Score during the 5-day Period	Num	Derived		8		PARCAT1='DIARY', PARCAT2='NRS SCORE', derived parameter showing the overall pain score per AVISIT, where PARAMCD in('EMSD01') as the average of the AVAL values. See SAP 6.2.1.
390	ADQSAH	PARAMCD	PLPNDM	Dysmenorrhea Pain Score during the 5-day Period	Num	Derived		8		PARCAT1='DIARY', PARCAT2='NRS SCORE', derived parameter showing the overall pain score per AVISIT, where PARAMCD in('EMSD01'), as the average of the AVAL values. See SAP 6.2.1.

Also for a non BDS domain, the link between domain, variable and derivation rule is present. But there is no link to TLF specifications. Therefore even though from TLF metadata we know which ADaM dataset is used in a table we do not know which analysis variable or which PARAMCD was used for which analysis result.

Following with our standard template for Protocol and SAP, we need to identify for each critical analysis result, mainly efficacy tables and/or figures the 'Reason', the 'Selection Criteria' and where in the file this information can be found. But our standard template for Protocol and SAP are, just like the TLF specification file, Word documents. So, even if we could identify the part of the standard that it is to be used in ARM, how could we read it in programmatically? Only standardised sections can be identified quickly but we cannot automate or relate the content as easy as with an Excel file.

However, in these files we can ensure vocabulary consistency across files. For example between Protocol, SAP, and TLF specification files, terminology used in each standard subsection should be determined based on the wording used in the 'predecessor' standard file.

The term 'predecessor' is used here as the standard file created before. Protocol is the predecessor of SAP and subsequently SAP is the predecessor of TLF specifications. A common example, the section called 'primary efficacy analysis' is used in SAP, while for the corresponding section in the protocol 'primary efficacy objectives' is used. Keeping vocabulary consistent across files, avoids misunderstanding and reduces the time used to familiarise oneself with the study.

Finally, some SAS code can be present in standard macros or programs, but not all programs would be standards. However, we could standardise a comment in the code to identify which part of the code can be used in ARM. And we have the location of this program in our TLF metadata, which is one step further. For non-standard macros or programs, we could instruct the programmers to add a 'comment' within the code for the piece of code that would be part of ARM. At this point having the critical or efficacy tables/figures marked in the TLF metadata would be helpful.

CREATING ADVANTAGES

Once we identify the elements of interest in each of the standard files we can modify the standards to facilitate the creation of ARM. Examples of possible changes will be shown in this section.

As mentioned above we cannot easily use the contents from a Word file (Protocol, SAP, TLF specs.). And creating an extra Excel file including the information from a Word file in a structured way does not seem an efficient solution. We would be creating extra work in another part of the process while the objective was decreasing the extra work needed to create ARM.

Some possible improvements from what we just saw in the previous section:

- Vocabulary used in interrelated standards documents like ADaM specifications, Protocol, SAP, TLF specifications, TLF metadata should be consistent. We could use the vocabulary from the CDISC ADaM Analysis Result Metadata (ARM.ANALYSIS RESULT, ARM.REASON; ARM.SELECTION CRITERIA), in the first predecessor file (e.g. Protocol).
- As an immediate simple improvement we could modify our TLF metadata as follows:

PhUSE 2016

- From Protocol and SAP, the list of TLFs included in ARM should be identified. Normally we include the TLF specification Table of Contents (TOC) in the TLF metadata. At this point TLFs part of ARM could be identified by the Statistician, either in the TOC of the TLF specification, or in the TLF metadata, and therefore marked as 'Y' in the extra column proposed 'ARM' in Figure 5.
- Adding an extra tab to include the link between Analysis result and the output (table, or figure). Because it is possible to have more than one analysis result per table/figure. The extra tab could be filled in by CP at the moment of programming the analysis result.
- The content of ARM can be QC-ed by the clinical programmer who is in charge of dual programming, if dual programming was used. In this case the TLF metadata fields, single & dual programmer, can be used.

Figure 5: TLF Metadata Improvements

Program name	Output Name	Output Type	Output Number	Title 3	Title 10	Footnote 2	Footnote 3	Footnote 9	Source	ARM	Program Location	Developer	Verifier	Comments
t_121121	_as	Table	12.1.1.2.1	Analysis Sets in REG1		{1} All randomized subjects who took at least one dose of study medication (DRUGA or DRUGB or placebo).	{2} All randomized subjects who received at least one dose of double-blind study medication and who have a primary NRS pain score for overall pain		ADSL	N	directory/directory/directory/program_name.sas	CPA	CPB	
example1	1	Table	12.3.1.1	Mean NRS Pain Score Over the Last 5 Days for Overall Pain. Inferential Analysis		Includes all treatment groups and information until Visit 6 (included).	Last non-missing observation before first intake of study drug is used as baseline.		ADQS	Y	directory/directory/directory/program_name.sas	CPA	CPB	
example1	2	Table	12.3.1.2	Mean NRS Pain Score Over the Last 5 Days for Overall Pain. Inferential Analysis		Only includes subjects randomized to placebo in Part 1 and taking DRUGA in Part 2. Columns refer to DRUGA treatment from Visit 6 onwards.	Last non-missing observation before first dose of DRUGA is used as baseline.		ADQS	Y	directory/directory/directory/program_name.sas	CPA	CPB	
example1	3	Table	12.3.1.3	Mean NRS Pain Score Over the Last 5 Days for Overall Pain. Inferential Analysis		Includes subjects within the same treatment group during the whole study and information for all visits.	Last non-missing observation before first intake of study drug is used as baseline.		ADQS	Y	directory/directory/directory/program_name.sas	CPA	CPB	

- Although there is no difficulty for the creation of ARM, this metadata is something relatively new and therefore it can be scary. Training including practical examples of Analysis Result Metadata should be available for statisticians and clinical programmers.

After a simple review of our available standards and some small improvements identified we are still missing some important connections. Links that needs to be done mentally by the clinical programmer at the moment of programming the analysis result, or by the statistician at the moment of writing the SAP or TLF specifications. Shall we then check when in the process of creation of each standard, the implicated sections are created? And see how we can improve the creation of ARM by inputting this information at the creation of the standard file involved?

GAIN SPEED AND EFFICIENCY

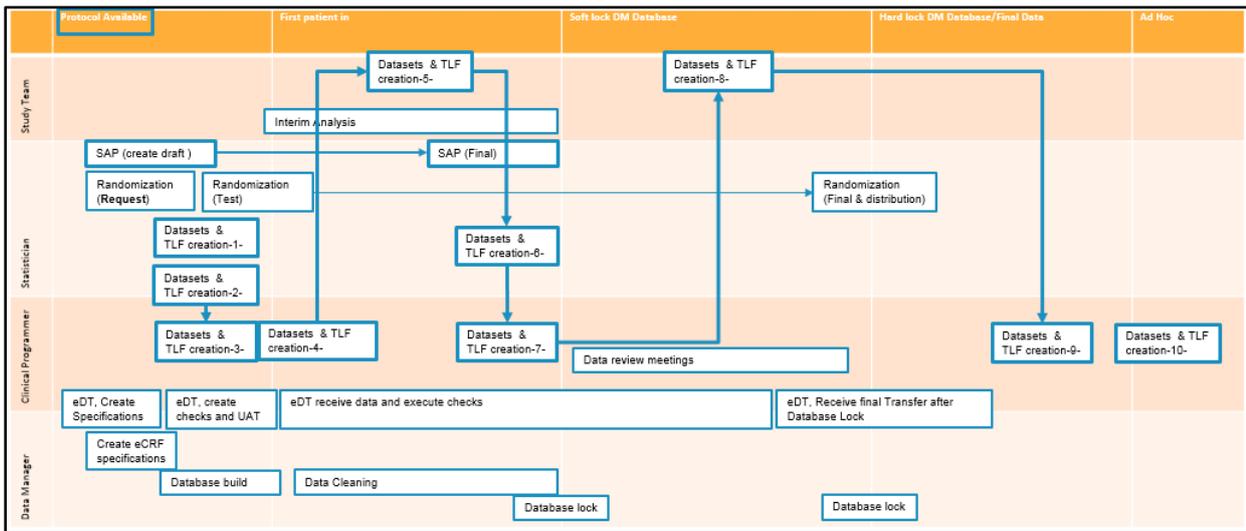
Most efficiency is won by inputting the correct information once and at the most opportune time. This section includes diagrams that show which relationships we can create between standard files and when in the process of TLF creation this information should be entered.

Note that the example below is a simplified version of the process that our sponsor follows, it may be different in other companies.

If we show the study life chronologically, we could simplify the TLF creation processes as shown in the following diagram (figure 6).

As we mentioned before we are only interested in the standard files that are a source of ARM, therefore our diagram can be simplified even more by only focussing on the sources for ARM, see only highlighted fields below.

Figure 6: Study Life - Flow Chart



The first standard file is the **Protocol**, it is created by clinical writers at the beginning with the participation of medical and statisticians, and the study team can participate in its review. At this point of the process Statisticians and Clinical programmers can provide comments during the Protocol review, especially on the section 'trial objectives' or at each 'endpoint', that can help to highlight the information needed in ARM later on in the process.

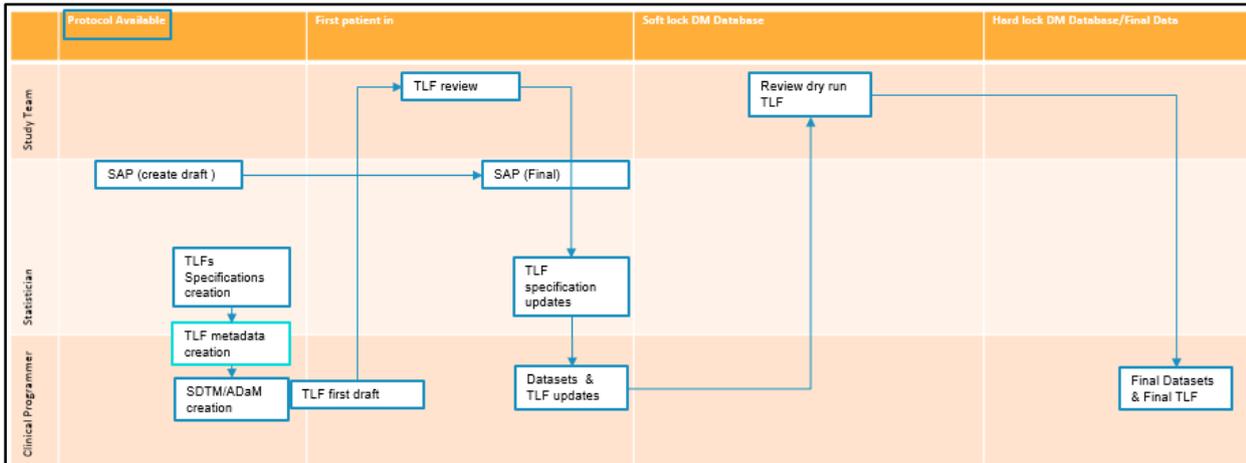
The next standard file, the **SAP**, is created after the Protocol is available and therefore can keep vocabulary consistent. The standard sections in the SAP, like 'Efficacy Endpoints' or 'Primary Efficacy', 'Secondary Efficacy Endpoints' or 'Key Secondary Efficacy', 'Safety variables', 'Analysis of safety', 'Analysis of Efficacy' sections should keep consistent vocabulary as in the protocol.

In our case **TLF specifications** are created almost in parallel with the SAP. Normally the same statistician is in charge of SAP and also TLF specifications creation, therefore here vocabulary consistency can be kept.

TLF specifications and TLF metadata: In our diagram there are several activities grouped by 'Datasets & TLF creation'. As first step the statistician creates TLFs Specifications (*1 in Figure 6). Based on the TLF specification table of contents (TOC) the statistician together with the clinical programmer create the TLF metadata (*2). After this step the clinical programmer creates aCRF, SDTM and ADaM specifications and datasets (*3). After first patient in and before soft lock the Clinical programmer creates TLFs first draft (*4). The TLFs drafts are reviewed by the study team (*5). And updates to TLF specifications are made by statistician in the next step (*6), here some modifications can affect the ARM metadata. CP updates and resolve QC findings and develops TLFs for Dry Runs (*7), between soft lock and hard lock the study team review the dry run TLF (*8) and finally CP generates final Datasets and TLF outputs (*9). In case ad-hoc TLFs are requested, these are produced by CP (*10) and reviewed by the study team.

Looking at the activities as detailed above, the best point in the process to start creating Analysis Result Metadata is at the time of TLF metadata creation, see Figure 7.

Figure 7: Study Life – Datasets & TLF Detailed Activities



The TLF metadata is a tool created after Protocol, SAP and TLF specifications are almost final. It is also a tool that is developed by CP and Statistician, both have access to it and can modify it during the TLF production. And it is used to track the status of the analysis output, which is related to each analysis result in ARM.

We could implement the simple changes suggested in previous section (Figure 5), by including in our existent TLF metadata, an extra Excel sheet like the example below. The list of TLFs included in ARM could then be identified in this TLF Metadata Excel tab.

Figure 6: TLF Metadata Modified for ARM

Program name	Output Name	DISPLAY IDENTIFIER	Develop	Verifier	DATASET	PARAMCD (CP)	ANALYSIS VARIABLE (CP)	RESULT IDENTIFIER (STAT)	REASON (STAT)	DOCUMENTATION (STAT)	SELECTION CRITERIA (CP)	PROGRAMMING STATEMENTS (CP)	Status
example 1	1	Table 12.3.1.1	CPA	CPB	ADQS	FLPNOV	PCHS	Treatment difference results (Mean, confidence interval, p-value)	Primary efficacy analysis as pre-specified in protocol	See Protocol Section XX for details. Program: program_x1.sas. MRS scores were analysed using an ANCOVA model which included dose group and region (REG1 and REG2) as fixed factors _	fasifl='Y', param='FLPNOV', trt01pn='-', avis='EoT'	proc mixed data=pain; class trt georegn; model chg=base trt georegn; lsmeans trt/cl_	Not done Passed QC QC findin
example 1	1	Table 12.3.1.1	CPA	CPB	ADQS			Pairwise comparison...					
example 1	2	Table 12.3.1.2	CPA	CPB	ADQS								
example 1	3	Table 12.3.1.3	CPA	CPB	ADQS								

Using Excel we could take from the existent TLF Metadata the fields that uniquely identify one analysis output (table, listing, or figure) like the ones highlighted in yellow in the figure above (Figure 8), and link them with a unique analysis result. The rest of the fields are ARM fields ready for the statistician or Clinical Programmer to be filled.

Using an existing tool that people are used to and that is, moreover, available at the right time seems to be the most efficient way to fill this gap in linking everything together.

Once TLF metadata 'modified for ARM' is ready it can be used by either clinical programmers or statisticians. Some ARM fields like 'Reason' or 'Documentation' can better be entered by the statistician at the time of finalising the SAP. Whereas others like ARM fields 'Programming Statement' or 'Selection criteria' can easily be filled by the clinical programmer in charge of the particular table or figure, at the time of programming.

Please note that with the above suggested changes to TLFs Metadata, the creation of ARM can still have modifications until TLFs are final. For example for the field 'Programming Statement', this is done by cutting and pasting the important PROC statements and selection of records from each SAS code. If this is done too early, the program code can change and programmer may forget to update this section. Therefore it is assumed that a QC for the final version of each analysis result is done and that the SAS code related to it was modified before final QC.

WHAT IS NEXT

The future, pressing a button to create most of the standard TLFs. As a programmer this might look scary but let's hope that statisticians will keep being creative with non-standards and challenging TLFs. We all try to avoid repetition if we can then why not have standard TLFs programming been automated?

There are other articles that show the path from Analysis Result Metadata to the generation of automated macros, this topic is not covered in this paper.

PhUSE 2016

CONCLUSION

We still have a long way to go, and we might need to revise our steps in the process. Let's stop for a moment and have a closer look at our existing tools and standard files, let's use this metadata to think further.

Creating this metadata only when the submission decision is taken, has a big disadvantage of having limited time. Having ARM ready for all studies as part of standard study work can avoid stressful situations. Also for actual studies that will become 'legacy' studies in the future, having ARM ready could simplify future submission work.

ARM is created manually at the end of the process and most probably by a different clinical programmer. At this point it may be too late to get all relevant information, moreover it often implies significant re work. Creating ARM is not impossible or extremely difficult but it takes time, especially with a lack of previous participation in the programming of the analysis results implicated.

By identifying possible improvements in our existent tools and standards, we could ensure that ARM includes all relevant information and avoid re work. As an obvious limitation in an intent of automatization we mentioned that Word type standard files are difficult to work with. However we still think there is place for improvements, see below what was the focus of this article:

- Within the organization, possible relationships between existent standard files that could facilitate the creation of ARM were identified
- In each standard source file for ARM the following was looked at:
 - o Who is the author and when is the standard file created
 - o How can standard files be related
 - o Possible additions in existent standard files

ARM training, or sharing practical examples, will improve communication between CP and STAT regarding Analysis Result Metadata creation or content.

A review of the complete process, what is created when and by whom can help to identify who would be most suitable during the different stages of a study for the completion of ARM and for the reviewing/QCing of ARM.

We should think in ARM as an internal needed information to have a good understanding of the critical tables instead of taking it as an obligatory submission document that will be needed for studies starting in March 15, 2018!

REFERENCES

- 1- <https://www.federalregister.gov/articles/2016/03/17/2016-05958/electronic-study-data-submission-data-standards-support-end-date-for-case-report-tabulation-data>
- 2- EMA [European Medical Agency] guidelines on nociceptive pain, CHMP/EWP/612/00, and on neuropathic pain, CHMP/EWP/252/03 and consensus from the medical scientific literature [Vincent et al, 2005] state using a 11-point NRS anchored by '0 = no pain' and '10 = worst pain you can imagine' as a validated and preferred instrument to document pain.
- 3- Analysis Results Metadata v1.0 for Define-XML v2.pdf from <http://www.cdisc.org/standards/foundational/adam>

CONTACT INFORMATION

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

Carla Santillan
Chiltern International bvba
Greenhill Campus, Interleuvenlaan 15E,
3001 Heverlee, Belgium
Work Phone: +32 492 59 36 03
Fax: +32 16 39 11 00
Email: carla.santillan@chiltern.com
Web: www.chiltern.com

Brand and product names are trademarks of their respective companies.