

ADaM Development Process, Template, and Experience

FMD K&L, Inc.

Xin Ke, 柯欣

November 28, 2014

Content

1

- General Thinking About ADaM

2

- ADaM Generation

3

- Template

4

- Experience, Common Errors

General Thinking of ADaM Generation

◇ Things to Think About When Starting ADaM

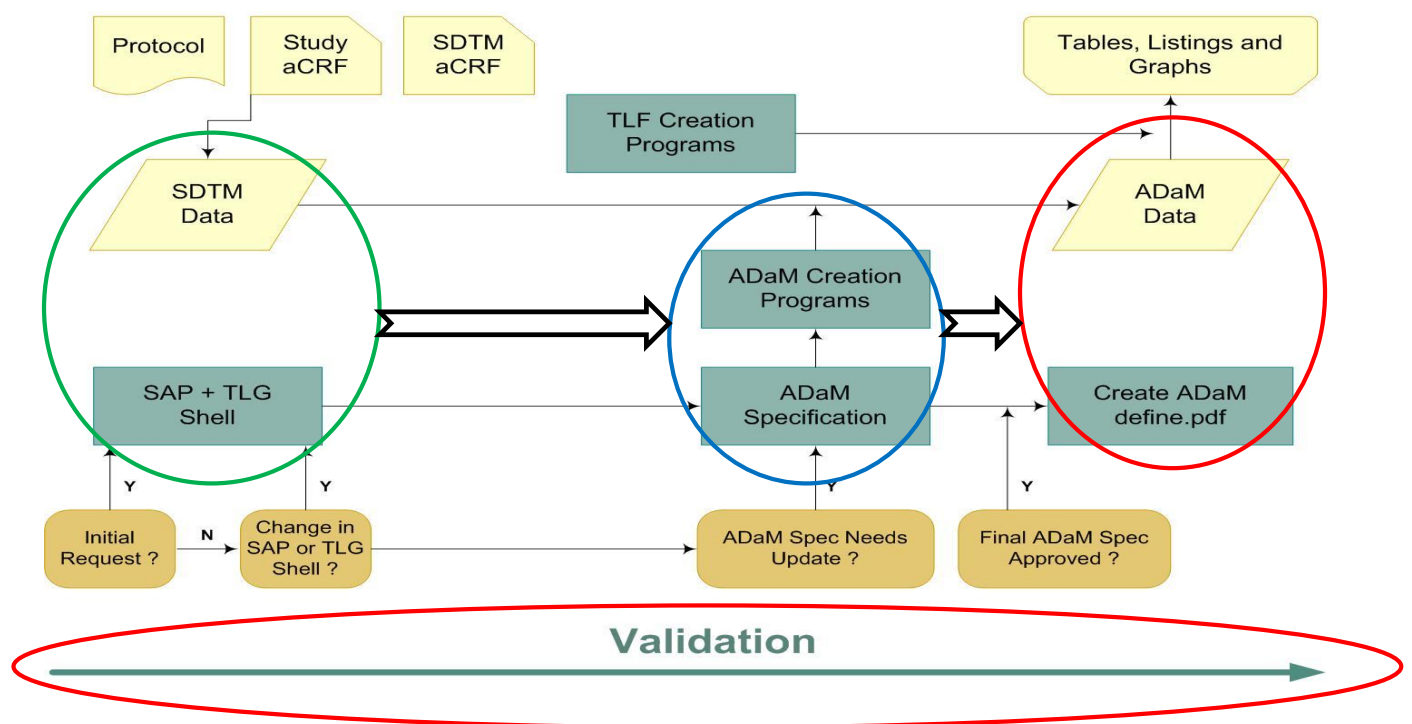
- Protocol/SAP/TLF Shell
- What to support – Single study or ISS/ISE, Submission?
Ad hoc planned?
- Integration – Involving integration and/or consistency from previous data or data created by others
- Source – Annotated CRF/SDTM, other external data.
Un-blinding

General Thinking of ADaM Generation

- Coding – dictionary version, lab (and others) unit, controlled term
- Traceability – Evaluate feasibility to build traceability
- Spec Template, standard programs
- Validation strategy, compliance check
- Submission package – define.xml, ADRG.pdf (ADaM reviewers guide) document, relation and timing of this part with others
- Team and assignment – ADSL, safety, efficacy

ADaM Creation Process Flow Chart

ADaM Data and TLF Flow Chart



ADaM Working Environment

- ◇ Standard folder structure
- ◇ Utility macro system for ADaM creation (list of samples)
 - %setup – initial environment setup
 - %sdtmplus – combine main domain and SUPPQUAL to make SDTM+ data
 - %adamspec – read ADaM spec in Excel into temporary SAS dataset
 - %adambuilder – create final ADaM dataset(s) by applying data and variable attribution from ADaM spec and SDTM domains
 - %splitvar – split variables when text is more than 200 characters
 - %splitsize – control ADaM dataset size by input parameters
 - %lengthadj – reduce value length as to be needed

ADaM Working Environment

- ◇ Utility macro for error checking (list of samples)
 - %chkSource – check data source are correctly listed
 - %chkKeyVar – check if key variables can uniquely identify single record
 - %chkTrunc – check if there is potential truncation
- ◇ Functional macros

ADaM Generations

- ◇ Specification development
- ◇ Statistician review, specification sign off
- ◇ Programming development
- ◇ Error checks and validation
- ◇ Compliance check and resolution
- ◇ Define.xml generation, ADRG.pdf generation
- ◇ Interim execution, specs/programming update life cycle, validation update, database lock, documentation
- ◇ Submission or final delivering package, QA approval

ADaM Template

- ◇ Used for ADaM specification development and dataset creation, define.xml generation, and communication platform during the whole development process
- ◇ Include all required information with some additional features
 - SDTM variable flag – indicate the variable source is in SDTM
 - Common variable flag in ADSL – indicate variable being copied to all ADaM datasets
 - Controlled Terminology – use CT name in Spec, and data driven process for entries
 - Derivation/comments – two types of comments: for stat/programmer and for reviewer/define.xml
 - Links – indicate the link to external file when algorithm is complex and needs external documental support
 - CodeList – list of controlled terms for reference and cross checks

ADaM Template Example – Contents Page

ADaM Dataset Specifications

Protocol	XX-YYY							
ADaM Version	0.1							
Dataset Name	Dataset Description	Dataset Structure	Program Name	Creation Order	Source Data	Key Variables of Dataset	Sorting Variables of Dataset	Class
ADSL	Subject-Level Analysis Dataset	One record per subject	adsl.sas	1	DM, DS, VS	USUBJID	USUBJID	ADSL
ADAE	Adverse Events Analysis Dataset	one record per subject per each AE recorded in SDTM AE domain	adae.sas	2	ADSL, AE	USUBJID, AETERM, AESTDT, AEENDT	USUBJID, AESTDTC, AESEQ	ADAE
ADVS	Vital Signs Analysis Dataset	One record per subject per parameter per analysis visit per analysis timepoint	adv.sas	2	ADSL, VS	USUBJID, PARAMCD, AVISIT, ATPT	USUBJID, PARAMCD, AVISIT, ATPT	BDS
ADTTE	Data for the Time to Event Analyses	one record per subject per parameter	adtte.sas	2	ADSL, YY	USUBJID, PARAMCD	USUBJID, PARAMCD	BDS
ADXX	<free text> Analysis Dataset	One or more records per subject per <analysis parameter> per <analysis timepoint>	adxx.sas	2	ADSL, XX	USUBJID, PARAMCD, AVISIT, ATPT	USUBJID, PARAMCD, AVISIT, ATPT	BDS
Code listing								

ADaM Template Example – ADSL Page (1)

Variable	Label	Type	Length	Display Format	Controlled Term	Core	Source	Derivation/Comments	Common Variable	Note to Programmer	Statistician Comment
STUDYID	Study Identifier	Char	20			Req	DM		Y		
USUBJID	Unique Subject Identifier	Char	30			Req	DM		Y		
SUBJID	Subject Identifier for the Study	Char	8			Req	DM		Y		
SITEID	Study Site Identifier	Char	4			Req	DM		Y		
COUNTRY	Country	Char	3	ISO3166	COUNTRY	Perm	DM				
INVID	Investigator Identifier	Char	10			Perm	DM				
INVNAM	Investigator Name	Char	100			Perm	DM				
BRTHDTC	Date/Time of Birth	Char	20	ISO8601		Perm	DM				
BRTHDT	Date of Birth	Num	8	YYMMDD10.		Perm	DM				
AGE	Age	Num	8			Req	DM		Y		
AGEU	Age Units	Char	6		AGEU	Req	DM		Y		
AGEGR1	Pooled Age Group 1	Char	50		AGEGRP	Perm	Derived	if . < AGE < 18 then AGEGR1 = '< 18 years'; else if 18 <= AGE <= 64 then AGEGR1 = '18 - 64 years'.	Y		
AGEGR1N	Pooled Age Group 1 (N)	Num	8		AGEGRPN	Perm	Derived	Numeric version of AGEGR1. See codelist AGEGRPN for the decode of each age group.	Y		

ADaM Template Example – ADSL Page (2)

Variable	Label	Type	Length	Display Format	Controlled Term	Core	Source	Derivation/Comments	Common Variable	Note to Programmer	Statistician Comment
SEX	Sex	Char	1		SEX	Req	DM		Y		
RACE	Race	Char	100		RACE	Req	DM		Y		
RACEOTH	Race Other, Specify	Char	200			Perm	DM		Y		
RACEGR1	Pooled Race Group 1	Char	50		RACEGRP	Perm	Derived	'White' if RACE=WHITE', Otherwise 'Non-White'.	Y		
ETHNIC	Ethnicity	Char	25		ETHNIC	Perm	DM		Y		
WEIGHTBL	Baseline Weight (kg)	Num	8			Perm	Derived	Derive from VS.VSSTRESN where VSTESTCD='WEIGHT'. For subjects with non-missing first dose date/time: Last non-missing measurement on or before first dose date/time(TRTSDTM). For subjects with missing first dose date/time:	Y		

ADaM Template Example – ADSL Page (3)

Variable	Label	Type	Length	Display Format	Controlled Term	Core	Source	Derivation/Comments	Common Variable	Note to Programmer	Statistician Comment
ITTFL	Intent-To-Treat Population Flag	Char	1		NY	Cond	Derived	ITTFL = Y if a subject took any portion of study medication, ITTFL = N otherwise.	Y		
COMPLFL	Completers Population Flag	Char	1		NY	Cond	Derived	COMPLFL=Y if there is an observation in DS with DSCAT='DISPOSITION EVENT' and DSDECOD='COMPLETED'; otherwise COMPLFL=N.	Y		
RANDFL	Randomized Population Flag	Char	1		NY	Cond	Derived	RANDFL = Y if SUPPDM.QVAL is not missing where QNAM='RANDNUM'; RANDFL = N otherwise.	Y		
DTHDT	Date of Death	Num	8	YYMMDD10.		Perm	Derived	Date part of DM.DTHDTC			
DTHFL	Subject Death Flag	Char	1		YNULL	Cond	DM				

ADaM Template Example – Code List

TRTP		ADSL
TRT01P	TRT01PN	
DRUG 600 mg	1	
DRUG 300 mg	2	
Placebo	3	
TRTPN		ADSL
TRT01PN	TRT01P	
1	DRUG 600 mg	
2	DRUG 300 mg	
3	Placebo	
AGEGRP		ADSL
AGEGR1	AGEGR1N	
< 18 years	1	
18 - 64 years	2	
AGEGRPN		ADSL
AGEGR1N	AGEGR1	
1	< 18 years	
2	18 - 64 years	
RACEGRP		ADSL
RACEGR1	RACEGR1	
White	White	
Non-White	Non-White	

Traceability

◇ Reviewer uses traceability to

- trace sponsor's results back to the CRF data
- understand the relationship between the analysis results, analysis datasets, SDTM datasets, and source data
- determine the observations and algorithms used to derive variables
- understand how statistics calculation (e.g. p-value) was conducted

◇ Traceability is a major issue in FDA submission

- Most common problem for submitted data per FDA
- Common problem for legacy data or ADaM derived independently from SDTM generation
- Caused delay for reviewing process, and questions to sponsors

Traceability

- ◇ Build traceability when developing ADaM specs and datasets
 - Metadata traceability – description how AVAL/AVALC is derived through other variables of domains – by (mathematical) formula
 - Data point traceability – indicate the immediately source data record of the current record. Commonly use SDTM domain name, SDTM variable name, and SDTM xxSEQ variables
 - ADaM variables SRCDOM, SRCVAR, and SRCSEQ – specially designed for the purpose of data point traceability to link AVAL/AVALC to specific SDTM domain, variable, and record
 - DTYPE is used to identify derived row and its associated algorithm: LOCF, WOCF, LVPD, LOV, MAXIMUM, MINIMUM, AVERAGE, ENDPOINT
 - Sponsor defined: e.g.: (CRITREL, CRITFACT) were used to trace source from multiple rows

Several Small Things Requiring Big Attention

◇ Split

- Should not be larger than 1GB or splitting is needed
- Split by Category, not by site/subject
- Resize the dataset to the maximum length used for each character variable prior to splitting
- SDTM variables will still need to be kept the same length as in the SDTM package
- Original data (large size) should still be included. Create subfolder \SPLIT under \adam, and included all split data under this subfolder
- Clearly name the split datasets to aid the reviewer in reconstructing the original dataset (e.g., adlb1, adlb2, ...)
- Define.xml and ADRG.pdf should clearly document and specify how the dataset is split

◇ Traceability

- Built-in traceability especially for key analysis is essential

Several Small Things Requiring Big Attention

◇ Date Variable

- ADaM date and time variable needs to be in numerical format so can be used directly for calculation

◇ USUBJID and SUBJID

- One subject can only have a single unique identifier (USUBJID) across the entire submission if the subject enrolled to multiple studies
- An individual subject should have the exact same unique identifier across all datasets, including between SDTM and ADaM datasets
- SUBJID should mean the same thing across studies

◇ Define.xml and ADRG.pdf

- Define.xml needs to be checked to be compliant with CDISC standard. Sometimes Sponsor did not run compliance check and OPENCDISC fires errors when package submitted to the agency
- Missing ADRG.pdf document – it is also required now

Examples of Some Challenges

◇ BDS

- BDS has fixed structure (vertical) with fixed set of variables. Preference is not to add any additional new variables. Many goals were achieved by structuring: CRITy/CRITyFL/CRITyFN and ANLzzFL/ANLzzFN

◇ ADLB

- Need to handle both conventional unit and SI while keeping PARAM/PARAMCD as 1-1 match
- Getting much more complicated because of other derived rows if keeping both unit systems in one dataset
- Decided to create ADLB and ADLBC for hosting SI unit only and conventional unit in separate dataset

Examples of Some Challenges

◇ ADSF36

- Questionnaire SF36 observed dataset
- Need many derived scores for analysis
- Create a separate analysis dataset ADSF including analysis records only

◇ Traceability

- AVAL in a row is derived multiple rows in the same dataset
- Used CRITREL and CRITFACT for this purpose

◇ When doing the design: to be comprehensive vs Simplicity

- Choose simplicity over comprehension

Questions and Comments

