

VALIDATE METADATA (define-xml)

What makes a define-xml conformant?

- XML code must be syntactically correct
- Valid according to the associated XML Schema
- Metadata vs dataset attributes and content
- Appropriateness of comments and methods

For example in Pinnacle21

- Validate define-xml alone**→ Make sure your define-xml is compliant with CDISC define-xml standard
- Validate datasets with define-xml** → Make sure what you have planned (define-xml) is what you have done (datasets)
 - Any missing variable in define-xml?
 - Any missing variable in the datasets?
 - Variables attributes match define-xml attributes?
 - Any occurred term that is not described in the define-xml?

THE PROBLEM

How to guarantee traceability between SDTM and ADaM?



- SDTM Only:** source variables and records not used in ADaM i.e. Screen Failures, IE domain, Suppl. Lab. Data, etc.
- SDTM and ADaM:** variables and records copied from SDTM to ADaM → **Traceability Issue 1**
- ADaM Only:** derived or assigned variables / new records in ADaM for analysis purpose → **Traceability Issue 2**

TRACEABILITY ISSUE 1

Check for ADaM Origin=Predecessor

- Keep all variables attributes from SDTM
- Bring codelist defined in SDTM
- Content must be not changed

The same applies when Predecessor is another ADaM

Easy to check programmatically

TRACEABILITY ISSUE 2

- Check for Clear Pattern of Derivations
- Creation of Records if needed e.g. impute missing observations
- Clear description of derivations

It requires more Independent and Manual Review

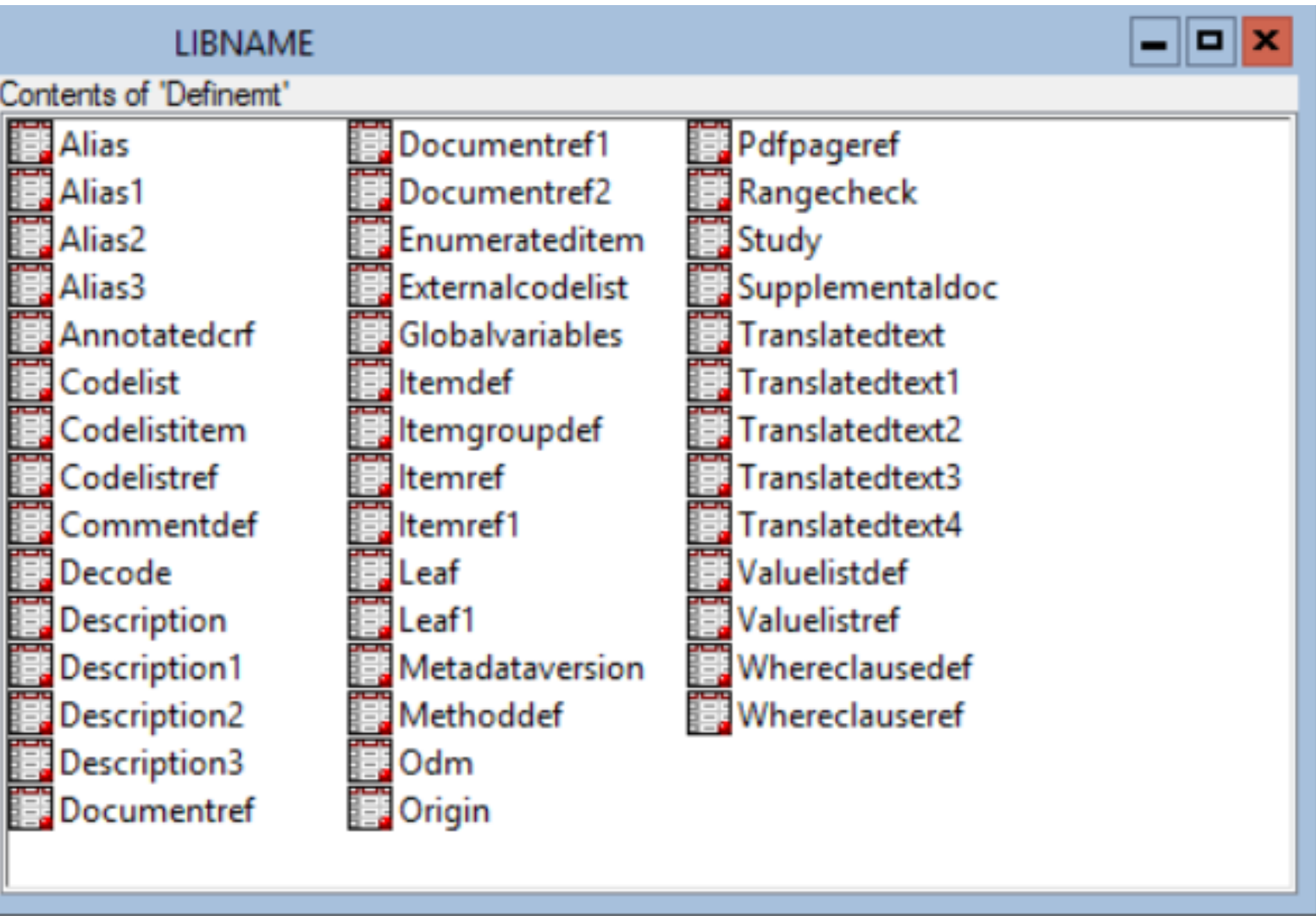
MORE DATA / METADATA DRIVEN, MORE

.... BUT

SOME GOOD INTERIM SOLUTION

CAPTURE DEFINE-XML METADATA IN SAS

```
filename define          "...\\define.xml" ;
filename map             "...\\Work\\defineXMLautomap.map";
libname define          xmlv2 automap=replace xmlmap=MAP;
libname defineMt         "...\\Work";
proc copy inlib=define outlib=defineMt;
run;
libname define clear;
```



Accessing the Metadata from Define XML ", Lex Jansen, PharmaSUG 2018

EXAMPLE OF AUTOMATIC SDTM / ADAM METADATA CHECK

Issues	Dataset	Variable	Label	Codelis	Origin	Predecessor
[ERROR: Variable does not exist in predecessor dataset]	ADAE	AEHLTGCD	Group Term Code	MEDDRANUM	Predecessor	AE.AEHLTGCD
[ERROR: Variable does not exist in predecessor dataset]	ADAE	AECAT	Category	AECAT	Predecessor	AE.AECAT

The two variables do not exist in the SDTM while the ADAM specifications are indicating Origin=Predecessor

EXAMPLE OF AUTOMATIC CHECK AND INHERITANCE OF METADATA FROM SDTM TO ADaM

Issues	Dataset	Variable	Label	Codelis	Origin	Predecessor
[NOTE: CT assigned from SDTM define]	ADMH	MHENRTPT	End Relative to Reference Time Point	ENRTPT	Predecessor	MH.MHENRTPT

Codelist was not assigned in the ADaM specifications, so our macro assigns it from the SDTM define.xml based on ADAM Predecessor metadata

ABSTRACT

Most of the companies **validate metadata** (define-xml) using a standard set of rules defined by CDISC and available in the "de-facto" standard tools such as Pinnacle21; the **validation is usually performed on individual packages, SDTM or ADaM**. However, these set of rules and the way they are implemented, lack of traceability checks from ADaM to SDTM.

With this presentation we would like to stress on the importance of **metadata and data consistency between SDTM and ADaM and between ADaM datasets** and how at Cytel we were able to fill the gap with a number of internal SAS standard programs and macros.

For example, if an ADAM variable has origin=predecessor the variable exist in SDTM and with same attributes, or if a variable from ADSL is replicated into other ADaMs the attributes and content is not changed. Moreover, the tools re-use SDTM metadata when a variable in ADaM define has origin=Predecessor.



Angelo Tinazzi, Senior Director, Standards, Systems, CDISC Consulting Statistical Programming, Cytel Inc. angelo.tinazzi@cytel.com

- >20 Years of Experience
- Passion for Standards
- Member of CDISC-EU Committee
- CDISC Authorized Instructor for ADaM

Subscribe to Cytel Blog to receive update on my **Good Data Submission Doctor** Blog Series <https://www.cytel.com/blog/topic/cdisc>