

## **Find / Track / Check and Close, Using SAS to Streamline SDTM Validation Including the Hyperlinks**

David Tillery, Smith Hanley Consulting, Lake Mary, Florida USA

Qiang Zhai, Purdue Pharma L.P., Stamford, Connecticut USA

Lily Peng, Purdue Pharma L.P., Stamford, Connecticut USA

### **ABSTRACT**

A number of vendor developed systems for SDTM compliance have been available for some time. While coding to identify SDTM validation issues is a straightforward process, methods of reporting, tracking, querying, resolving and closing items remains both challenging and time consuming, especially for certain manual processes. Even by electronically sourcing every CDISC standard element including SDTM domains, controlled terminology, annotations and hyperlink destinations, without a dynamic tracking interface, resolution remains tedious. Using SAS® with a little help from some drop-down Acrobat® features, this paper will illustrate a three-module approach including basic compliance data issues as well as the prevalent define.xml linking issue. A well designed system eliminates queries to the data, as well as the PDF and XML “click and hope” validation method.

### **INTRODUCTION**

A data transfer triggers the validation effort. Statistical Programming takes a total package review approach with three basic modules:

1. SDTM compliance including sponsor specific rules,
2. Electronic validation of CRF annotations,
3. The complete define.xml “dismantling” including origin hyperlinks.

The first module includes adopted control terminology for both extensible and non-extensible lists.

The second module validates the extensive annotation rules developed with years of experience with a basic guidance from CDISC. Electronic validation values for hyperlinks are extracted as well.

The third module presents the entire define.xml, element by element, attribute by attribute, in a manner which mimics the input layout, assuring an unambiguous review cycle.

By adopting this approach, two critical equivalencies are communicated to our Providers:

- Understanding and reporting the metadata is equally important to the data itself
- Reporting and tracking an issue is equally important to the identification of the issue itself

## MODULE 1 – SDTM COMPLIANCE

### Methodology – generate issues only

The simple rule for compliance items, like a non-extensible list value within a variable, is checked with some basic SAS programs. Issues are always reported to a “just enough” level of detail. A typical issue generated from the SAS program would be reported as

Controlled Terminology - DOMAIN="AE" VARNAME="AEACN" has a value="" occurring 4 times not matching accepted CT values

The study Data Manager or Study Programmer never needs to check the AE dataset, only report the issue. With this, the CRO knows exactly “what, where, and how many.”

The reporting and resolution structure cycle is strictly controlled. Controlled Terminology for the issue “**status**” column is as follows:

In-House Reporting use only	CRO use only
<b>Open</b> – program generated as issues are identified, items set to “Fixed” within a review cycle will be reset to “Open” if the validation does not support the “Fixed” status	<b>Open</b> – Not available
<b>Under Review</b> – placed on an issue, generally a controlled terminology extensible item which is being evaluating	<b>Under Review</b> – placed on an issue if resolution is not expected before the next transfer
<b>Fixed</b> – Not available	<b>Fixed</b> – indicates underlying data issue will not appear in the next transfer
<b>Closed</b> – program generated only if supported by the program review	<b>Closed</b> – Not available
<b>Request Forgive</b> – Not available	<b>Request Forgive</b> – used when the expectation is that the issue will remain through database lock
<b>Forgive Granted</b> – possible use with supporting documentation, or perhaps an instruction for manual or hard code intervention (strictly controlled with approvals)	<b>Forgive Granted</b> – Not available

Another example where more detail is reported:

SUPPAE – USUBJID <study>-xxxx-yyyy\*\*\* in AE does not have an expected QNAM=AEHLGT, IDVARVAL = <missing> - 2 records

This example indicates 2 AE records for a particular subject are not coded to the high level term.

This might be an example of an item the programmer would set to “Under Review” while the coding process catches up. “Under Review” would temporarily suspend the normal review process within the SAS macros, but would not be allowed at or near database lock.

Our SAS programs will generate the initial issue log at the agreed data transfer point, say 50%, with reasonable expectations of expected data issues.

Subsequent transfers require an updated issue log from the CRO programmer with the transfer, which becomes the basis to compare and close issues marked as "Fixed," as well as the continuation of the review.

Finally at database lock, the issue log passes a final electronic evaluation and the results are made available to QC.

## MODULE 2 – ANNOTATIONS

### Methodology – display everything and filter issues only

Many hours have been invested in a comprehensive set of annotation rules making a 100% electronic review a welcome reality.

When we say we want the annotations in

*"12 point, red italics, opaque, no border, no fill color, within a top 1 inch and 3 sides ¼ inch margin, all caps, 2 level rules, SDTM variable level rules, STDM value level rules, SUPPQUAL rules, [ ] around text information annotations, typo free with locked properties, with special sauce, lettuce, cheese, pickles on a sesame seed bun,"*

that is what we expect. And we will electronically validate every one of those rules.

Data transfers include the annotated Case Report Form named blankcrf.pdf. The evaluation and issue log generation is performed in a similar manner as Module 1, using SAS to evaluate the Acrobat exported FDF file where attributes, positions and properties are readily available. An issue, in this cycle, is a violation of annotation rules, so the issue log will present a structured and friendly rendering of the annotations which are easily filtered for "Open" issues.

An FDF export object looks something like this:

```
7 0 obj
<</C[0.75 1.0 1.0]/Contents(SITEID)/CreationDate(D:20071024160353-05'00')
/DA(0 0 0 rg /Arial,BoldItalic 9 Tf)/DS(font: italic bold Arial 9.0pt;
text-align:left; color:#FF0000)/F 132/Page 0/style="font-size:9.0pt;
text-align:left;color:#FF0000;font-weight:bold;font-style:italic;font-
family:Arial;font-stretch:normal"><p dir="ltr">SITEID</p></body>)
/Rect[209.606 547.187 265.61 560.197] /Subj(Text Box)/Subtype/FreeText/Type/Annot>>
endobj
```

A SAS program generates the following display of FDF objects and identifies rule violations as shown.

status	Issue	crfpage	domain	varname	annotation_variable	locked	Origin
Open	This variable requires a 2 level annotation DM.SITEID	1	DM	SITEID	SITEID	Yes	CRF Page 1
		1	DM	INVID	DM.INVID	Yes	CRF Page 1
		1	DM	INVNAM	DM.INVNAM	Yes	CRF Page 1
		1	DS	DSDTC	DSDTC	Yes	CRF Pages 1, 2, 17, 124
		1	DM	BRTHDTC	DM.BRTHDTC	Yes	CRF Page 1

Thus without ever having to look at CRF page 1, this OUTPUT is provided and later compared at the next transfer and "Closed" if the item is fixed. The origin field is also derived, which is later merged with the define.xml to validate the hyperlinks.

## MODULE 3 – METADATA, AKA DEFINE.XML

### Methodology – display everything and filter issues only

There's a reason they chose XML, just as there is a reason they told you to build a baseball field in the middle of a corn field. The voice of Darth Vader, from Field of Dreams, sort of.

XML can be a challenging parsing exercise. It is not something particularly well-suited to SAS. Past papers have addressed this with dozens of approaches for the XML files. Here are some real results, which complete the review package.

Every single attribute is checked against standards and house rules, right down to the spelling of variable names used in computational methods. And we mean everything. An extensive SAS program exports the results into 5 Excel worksheets: domain\_level, variable\_level, valuelist, codelist and computational\_method. Not by accident, this display format replicates the input structure used by our CRO's to produce the XML file. SAS program generated issues are at the attribute level.

Typical results for a variable level would be reported like this (transposed for display purposes)

status	Open
issue	Comment - typo RFSDTC
replace_origin	
replace_comment	
domain	DM
varname	DMDY
label	Study Day of Collection
datatype	integer
length	8
role	TIMING
origin	DERIVED
significantdigits	
comment	Calculation: = DMDTC-RFSTDTC+1 if DMDTC is on or after RFSDTC, Calculation = DMDTC - RFSTDTC if DMDTC precedes RFSTDTC.
mandatory	No
codelist_oid	
value_listref	
computationmethodoid	

The SAS macro picks up this typo error as a mismatch to all known SDTM standard variables.

The final validation worksheet exported is appropriately named "origin\_link\_issues" with a typical display of

status	issue	domain	varname	xml_origin	crf_origin
		VS	VSORRES	CRF Pages 4, 5	CRF Pages 4, 5
		VS	VSORRESU	CRF Pages 4, 5	CRF Pages 4, 5
Open	origin mismatch - this variable annotation is not found on Page 4	VS	VSPOS	CRF Pages 4, 5	CRF Page 5

Neither the rendered define.xml nor the blankcrf.pdf need be opened to report this issue, but again the CRO knows exactly what requires attention.

## CONCLUSION

Let your programs do the work. Stop wasting time with queries to the data. Source every possible element and implement a well thought out review process. Cycle times will decrease and quality will improve.

## REFERENCES

### PharmaSUG2004 Paper CC02 Using SAS to Speed up Annotating Case Report Forms in PDF Format

Dirk Spruck, Covidence GmbH, Marburg, Germany

Monika Kawohl, Covidence GmbH, Marburg, Germany

## ACKNOWLEDGMENTS

The team extends thanks to the following individuals for supporting this project (all of Purdue Pharma, L.P.)

Catherine Munera for her leadership and guidance

Theresa Belotti for her Data Management requirements input and review

Lou Florio for his SDTM expertise

Shideh Khorasheh for her Project Management support

## CONTACT INFORMATION

Your comments and questions are most appreciated. Contact the principal author at:

David Tillery

Email: [tillery@mindspring.com](mailto:tillery@mindspring.com)

David is an independent consultant represented by Smith Hanley Consulting under contract at Purdue Pharma, L.P.

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 the respective companies.