



AD01 - Validating a CDISC Dataset at development time using Pinnacle 21 CLI via SAS

David Meek

Introduction

CDISC dataset validation issues are often caught too late in the study development cycle. A minor update at ADaM level prior to a Sponsor delivery of TLFs can require a programming update tracking back to SDTM resulting in updates for minor compliance issues, which then in turn require a time-consuming re-run of TLF code to ensure the correct run order of dependencies for both production and QC

Scope

Presentation will cover individual dataset validation using Pinnacle 21 Community® validator versions 2.2.0 and 3.0.1 in a Microsoft Windows® environment

We will not discuss in detail the interpretation of P21 validation checks

Important Validation Note

Individual dataset validation is intended as an extra check at programming development time

ALL study datasets should be validated together as a package with the define.xml in P21 during/end of a study to ensure cross-check validation for compliance

X Command

X command in SAS allows submission of arguments to the Windows OS via a DOS prompt from SAS

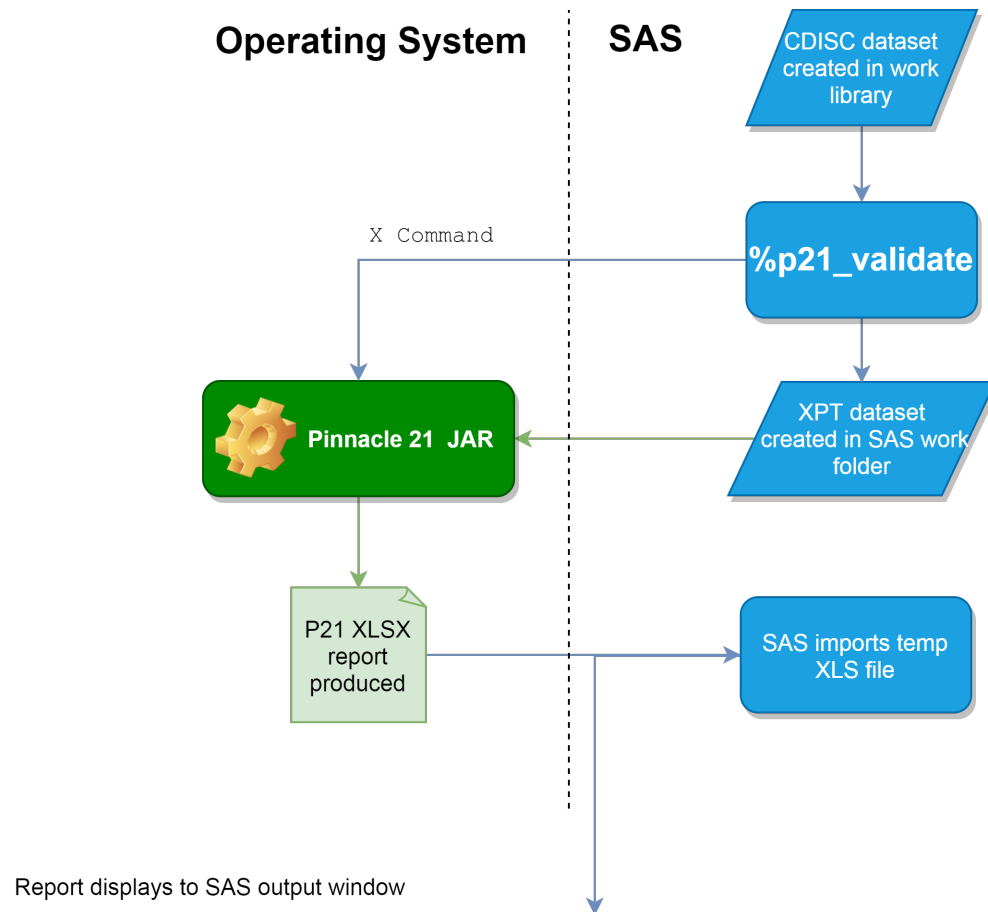
We can use SAS to submit arguments and execute the P21 validator application without the need for GUI

After validation we use SAS to import these results and provide real time validation results

More information on X command:

<http://support.sas.com/documentation/cdl/en/hostwin/63285/HTML/default/win-stmt-x.htm>

Process Overview



TestStudy EG P21 Validation Summary Report				
P21 ID	Publisher ID	Message	Severity	Count
SD1082	FDAB018	Variable length is too long for actual data	Error	13
CT2002	CG0021	EGTEST value not found in 'ECG Test Name' extensible codelist	Warning	175
CT2002	CG0021	EGTESTCD value not found in 'ECG Test Code' extensible codelist	Warning	175
CT2002	CG0021	EPOCH value not found in 'Epoch' extensible codelist	Warning	701
SD1339	FDAB022	Missing EPOCH value, when a start or observation date is provided	Warning	3

Creating a temporary folder in work area

Pinnacle 21 requires SAS XPT versions of the dataset

Need to create a sub-folder in the temporary SAS Work folder to save this XPT

Note: on closing of SAS this temporary folder gets deleted

```
*--- options to not hang on windows OS;
```

```
options noxsync noxwait;
```

**When creating folders,
we don't need SAS to
wait on OS**

```
*--- temp work path;
```

```
%let work_path = %sysfunc(getoption(work));
```

```
*--- temp subfolder to copy XPT to;
```

```
x mkdir "&work_path\xpt";
```

**Creates a sub-folder in
the temporary SAS**

```
*--- sleep SAS to allow the temp folder to be created;
```

```
data _null_;
```

```
  x=sleep(5);
```

```
run;
```

Creating a temp XPT version of the dataset

```
*--- remove sortedby to avoid warn-ing;  
data &domain. (sortedby=_null_);  
set &model..&domain.;  
run;
```

**Removes not SORTEDBY
warning with XPT files**

We then setup an XPORT library where we will create the XPT version of our dataset created by the PROC COPY:

```
libname temp_xpt xport "&work_path\xpt\&domain..xpt";  
  
proc copy in=work out=temp_xpt;  
select &domain.;  
run;
```

**Creates XPT version of
domain in sub-folder**

```
*--- options to not hang on windows OS;  
options xsync xwait;
```

**Forces SAS to wait on OS
process i.e. P21 validation**

P21 CLI 2.2.0 – X Command Code

```
%*--- single domain validation;
```

```
x java
```

```
-Xmx1024M -Xms1024M
```

```
-jar \\tools\pinnacle21-community-2-2-0\components\lib\validator-cli-2.1.5.jar
```

```
-type=&model
```

← SDTM or ADaM

```
-source="&work_path\xpt\&domain..xpt"
```

← Dataset to be validated

```
-source:type=sas
```

```
-config="\\tools\pinnacle21-community-2-2-0\components\config\&model. &model_version..xml"
```

```
-config:cdisc=&terminology
```

← NCI CT version i.e. 2019-06-28

```
-report=%sysfunc(quote(&work_path.\temp_val.xlsx))
```

```
-report:type=&type.
```

← Excel, CSV

← P21 report

```
-report:overwrite=yes
```

```
-report:cutoff=&cutoff.
```

← Cut-off for report obs
defaulted to 1000

Location of JAR
file used by CLI

SDTM 3.2,
ADaM 1.0

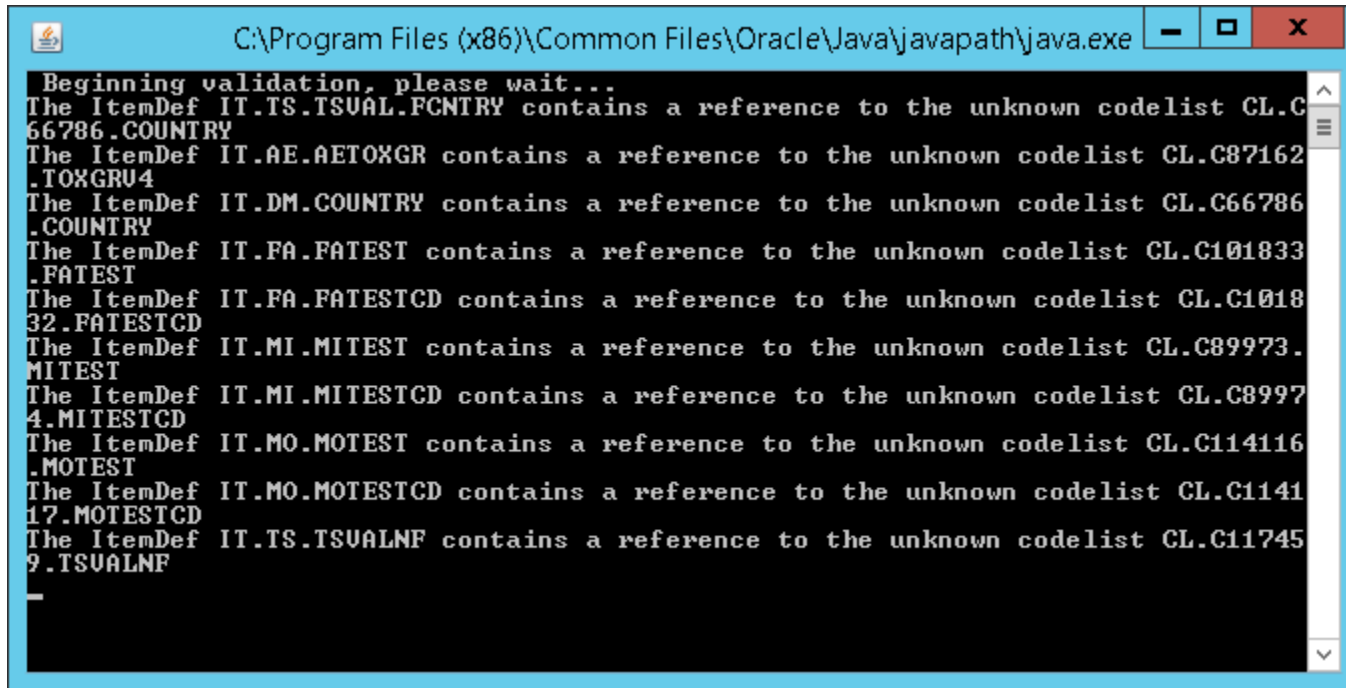
P21 CLI 2.2.0 – CLI

Example of resultant commands fed into the external P21 validator on running the macro:

```
java -Xmx1024M -Xms1024M -jar \\tools\pinnacle21-community-2-2-0\components\lib\validator-cli-2.1.5.jar -type=SDTM -source="\\SAS Temporary Files\xxxxxxx\xpt\EG.xpt" -source:type=sas -config="\\tools\pinnacle21-community-2-2-0\components\config\SDTM 3.2.xml" -config:cdisc=2018-12-21 -report="\\SAS Temporary Files\xxxxxxx\temp_val.xlsx" -report:type=Excel -report:overwrite=yes -report:cutoff=1000
```

P21 CLI 2.2.0 – Running JAR

SAS will break out to windows using X command and run P21



A screenshot of a Windows command prompt window. The title bar shows the path 'C:\Program Files (x86)\Common Files\Oracle\Java\javapath\java.exe'. The window contains the following text:

```
Beginning validation, please wait...
The ItemDef IT.TS.TSUAL.FCNTY contains a reference to the unknown codelist CL.C
66786.COUNTRY
The ItemDef IT.AE.AETOXGR contains a reference to the unknown codelist CL.C87162
.TOXGRU4
The ItemDef IT.DM.COUNTRY contains a reference to the unknown codelist CL.C66786
.COUNTRY
The ItemDef IT.FA.FATEST contains a reference to the unknown codelist CL.C101833
.FATEST
The ItemDef IT.FA.FATESTCD contains a reference to the unknown codelist CL.C1018
32.FATESTCD
The ItemDef IT.MI.MITEST contains a reference to the unknown codelist CL.C89973.
MITEST
The ItemDef IT.MI.MITESTCD contains a reference to the unknown codelist CL.C8997
4.MITESTCD
The ItemDef IT.MO.MOTEST contains a reference to the unknown codelist CL.C114116
.MOTEST
The ItemDef IT.MO.MOTESTCD contains a reference to the unknown codelist CL.C1141
17.MOTESTCD
The ItemDef IT.TS.TSUALNF contains a reference to the unknown codelist CL.C11745
9.TSUALNF
-
```

Use of `xsync` `xwait` options forces SAS to wait until the OS process has completed (P21 validation) allowing time for the report to be generated before reading the report into SAS

Note: depending on datasets size run time can be from secs to several minutes

P21 CLI 3.0.1 – X Command Code

```
%*--- need lowercased macro params for input;  
%let model2=%lowcase(&model);
```

```
%*--- single domain validation;
```

```
x java  
  -Xmx1024M -Xms1024M  
  %if &local.=N %then %do;  
    -jar "%Tools\pinnacle21-community-3-0-1\resources\  
    app.asar.unpacked\components\lib\p21-client-1.0.1.jar" ^  
  %end;  
  %else %do;  
    -jar "C:\Users\&sysuserid.\AppData\Local\Programs\pinnacle21-community\  
    resources\app.asar.unpacked\components\lib\p21-client-1.0.1.jar" ^  
  %end;  
  --engine.version="%engine." ^  
  --standard=&model. ^  
  --standard.version=&model_version. ^  
  --source.&model2.="%work_path\xpt\&domain..xpt" ^  
  -cdisc.ct.&model2..version=&terminology. ^  
  --report="C:\Users\&sysuserid.\Desktop\temp_val.xlsx";
```

Central JAR file
used by CLI

Local JAR file
used by CLI

1903.1 (FDA) / 1511.5 (PMDA) – new param for 3.0.x

SDTM or ADaM

3.2 (SDTM) / 1.1 (ADaM)

Dataset to be validated

NCI CT version i.e. 2019-06-28

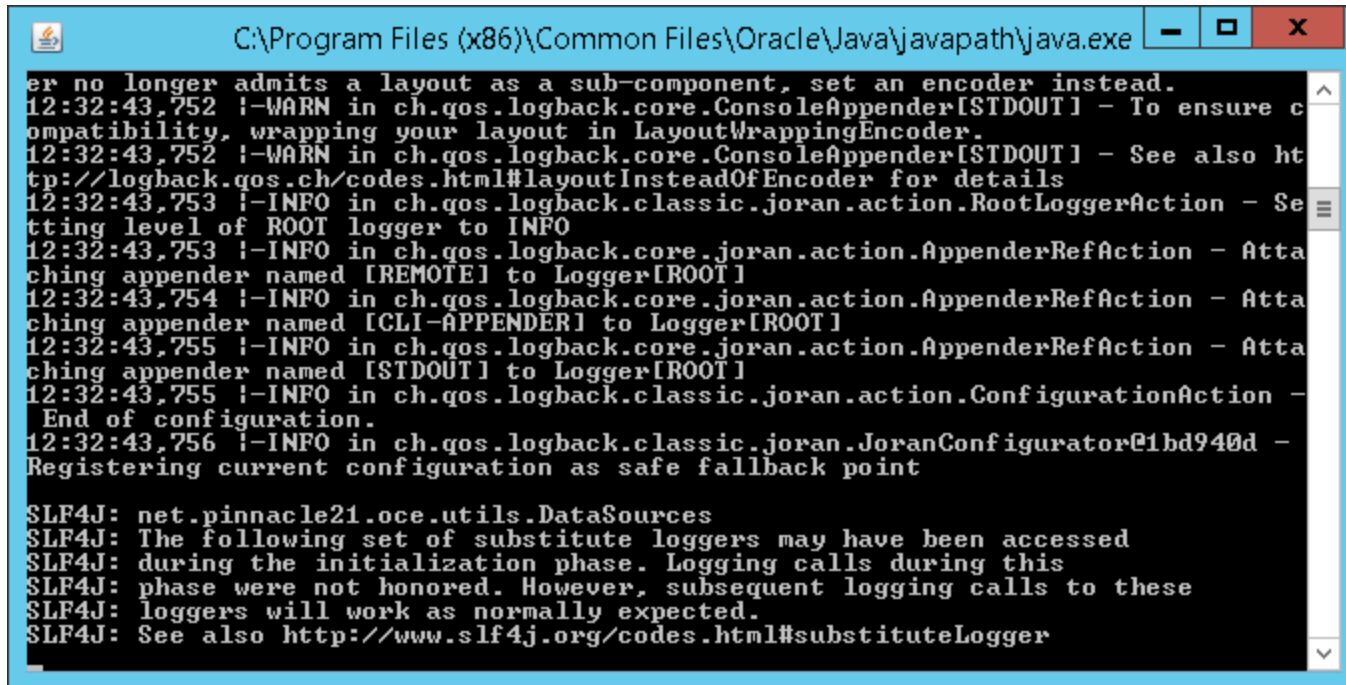
P21 report

P21 CLI 3.0.1 – CLI

Example resultant commands fed into the external P21 validator on running the macro:

```
java -Xmx1024M -Xms1024M -jar"\\tools\pinnacle21-community-3-0-1\resources\app.asar.unpacked\components\lib\p21-client-1.0.1.jar" ^ -  
-engine.version="FDA 1903.1" ^ --standard=SDTM ^ --  
standard.version=3.2 ^ --source.sdtm="\\SAS Temporary  
Files\xxxxxxx\xpt\EG.xpt" ^ -cdisc.ct.sdtm.version=2019-06-28 ^ --  
report="C:\Users\xxxxxxx\Desktop\temp_val.xlsx"
```

P21 CLI 3.0.1 – Running JAR



```
C:\Program Files (x86)\Common Files\Oracle\Java\javapath\java.exe
er no longer admits a layout as a sub-component, set an encoder instead.
12:32:43,752 !-WARN in ch.qos.logback.core.ConsoleAppender[STDOUT] - To ensure c
ompatibility, wrapping your layout in LayoutWrappingEncoder.
12:32:43,752 !-WARN in ch.qos.logback.core.ConsoleAppender[STDOUT] - See also ht
tp://logback.qos.ch/codes.html#layoutInsteadOfEncoder for details
12:32:43,753 !-INFO in ch.qos.logback.classic.joran.action.RootLoggerAction - Se
tting level of ROOT logger to INFO
12:32:43,753 !-INFO in ch.qos.logback.core.joran.action.AppenderRefAction - Atta
ching appender named [REMOTE] to Logger[ROOT]
12:32:43,754 !-INFO in ch.qos.logback.core.joran.action.AppenderRefAction - Atta
ching appender named [CLI-APPENDER] to Logger[ROOT]
12:32:43,755 !-INFO in ch.qos.logback.core.joran.action.AppenderRefAction - Atta
ching appender named [STDOUT] to Logger[ROOT]
12:32:43,755 !-INFO in ch.qos.logback.classic.joran.action.ConfigurationAction -
End of configuration.
12:32:43,756 !-INFO in ch.qos.logback.classic.joran.JoranConfigurator@c1bd940d -
Registering current configuration as safe fallback point

SLF4J: net.pinnacle21.oce.utils.DataSources
SLF4J: The following set of substitute loggers may have been accessed
SLF4J: during the initialization phase. Logging calls during this
SLF4J: phase were not honored. However, subsequent logging calls to these
SLF4J: loggers will work as normally expected.
SLF4J: See also http://www.slf4j.org/codes.html#substituteLogger
```

3.0.1 CLI requires an **active** internet connection for loading terminologies, dictionaries and rule implementations

Potentially poses an issue for servers that don't allow incoming connections

PHASTAR have found in practice 3.0.1 runs much slower than 2.2.0

Dataset Summary – EG Validated

Pinnacle 21 Validator Report

Configuration: \\tools\pinnacle21-community-2-2-0\components\config\SDTM 3.2.xml

Define.xml: Not provided

Generated: 2019-10-21T21:03:37

CDISC CT Version: 2018-12-21

Software Version: 2.1.2

**Report will be limited
compared to full
validation**

Processed Sources									
Domain	Label	Class	Source	Records	Rejects	Errors	Warnings	Notices	
GLOBAL	Global Metadata	--	--	--	0	2	8	0	
EG	ECG Test Results	FINDINGS	EG.xpt	878	0	13	1052	0	
Total				878	0	15	1060	0	

Unprocessed Sources									
Domain	Label	Class	Reason	Rejects	Errors	Warnings	Notices		
Total					0	0	0	0	

Grand Total				878	0	15	1060	0	
-------------	--	--	--	-----	---	----	------	---	--

Reading P21 Validation report into SAS

Some top level checks will generate by default when we validate an individual dataset i.e. “Missing DM dataset”

Our intention is to only validate a single dataset so this is expected

We can modify the report that is read into SAS to only keep what we need for reporting on the individual dataset being validated and other information can be dropped from the report we read back into SAS

As mentioned in the intro, datasets should always be validated together during a study and we are only looking at individual datasets for an extra check during development!

Summary Report – EG validation

Issue Summary					
Source	Pinnacle 21 ID	Publisher ID	Message	Severity	Found
GLOBAL					
	SD1020	FDAC001	Missing DM dataset	Error	1
	SD1115	FDAC002	Missing TS dataset	Error	1
	SD1106	FDAC005	Missing AE dataset	Warning	1
	SD1107	FDAC006	Missing LB dataset	Warning	1
	SD1108	FDAC007	Missing VS dataset	Warning	1
	SD1109	FDAC004	Missing EX dataset	Warning	1
	SD1110	FDAC003	Missing DS dataset	Warning	1
	SD1111	FDAC008	Missing SE dataset	Warning	1
	SD1112	FDAC009	Missing TA dataset	Warning	1
	SD1113	FDAC010	Missing TE dataset	Warning	1
EG					
	SD1082	FDAC036	Variable length is too long for actual data	Error	13
	CT2002	FDAC341	EGTEST value not found in 'ECG Test Name' extensible codelist	Warning	175
	CT2002	FDAC341	EGTESTCD value not found in 'ECG Test Code' extensible codelist	Warning	175
	CT2002	FDAC341	EPOCH value not found in 'Epoch' extensible codelist	Warning	701
	SD1076	FDAC031	Model permissible variable added into standard domain	Warning	1

Not relevant for single dataset validation, so can ignore and remove from report

Issues in single dataset specified for validation

Reading P21 Validation report into SAS

P21 validation report `temp_val.xlsx` generated is saved to the user's local desktop workspace:

```
*--- import validation checks - summary;
proc import datafile="C:\Users\&sysuserid.\Desktop\temp_val.xlsx"
    out=p21_summary
    dbms=xlsx
    replace;
    sheet="Issue Summary";
    getnames=yes;
run;

*--- import validation checks - summary;
proc import datafile="C:\Users\&sysuserid.\Desktop\temp_val.xlsx"
    out=p21_details
    dbms=xlsx
    replace;
    sheet="Details";
    getnames=yes;
run;

*--- delete temporary validation file as has been imported back into SAS;
x %unquote(%str(%'del "C:\Users\&sysuserid.\Desktop\temp_val.xlsx"%'));
```

Reading P21 Validation report into SAS

*---- keep validation checks specific to domain - drop general issues that arise due to incomplete validation of entire study;

```
data p21_summary;  
  set p21_summary;  
  retain pinnacle_21_validator_report2;  
  if pinnacle_21_validator_report ne "" then  
    validator_report2=pinnacle_21_validator_report;  
  if pinnacle_21_validator_report2="&domain." and d ne "" then output;  
  drop pinnacle_21_validator_report2;  
  rename b=p21_id  
  c=pub_id  
  d=message  
  e=severity  
  f=count;  
  keep b c d e f;  
run;
```



**Drop checks unless from dataset
we've selected for validation**

print_summary=Y (default=Y)

```
%if &print_summary.=Y %then %do;
title1 "&study. &domain P21 Validation Summary Report";
proc report data=p21_summary nowd headline headskip ls=150;
column P21_ID Pub_ID Message Severity Count;
define P21_ID / width=15 flow "P21 ID";
define Pub_ID / width=15 flow "Publisher
define message / width=70 flow "Message";
define severity / width=10 flow "Severity";
define count / width=10 flow "Count";

run;
%end;
```

TestStudy EG P21 Validation Summary Report

P21 ID	Publisher ID	Message	Severity	Count
SD1082	FDAB018	Variable length is too long for actual data	Error	13
CT2002	CG0021	EGTEST value not found in 'ECG Test Name' extensible codelist	Warning	175
CT2002	CG0021	EGTESTCD value not found in 'ECG Test Code' extensible codelist	Warning	175
CT2002	CG0021	EPOCH value not found in 'Epoch' extensible codelist	Warning	701
SD1339	FDAB022	Missing EPOCH value, when a start or observation date is provided	Warning	3

print_details=Y (default=N)

TestStudy EG P21 Validation Details

Record	Variables	Values	Pinnacle 21 ID	Publisher ID	Message	Category	Severity
	EGTEST	ECG Tests	CT2002	CG0021	EGTEST value not found in 'ECG Test Name' extensible codelist	Terminology	Warning
	EGTESTCD	EGALL	CT2002	CG0021	EGTESTCD value not found in 'ECG Test Code' extensible codelist	Terminology	Warning
	EPOCH	FIRST FOLLOW-UP	CT2002	CG0021	EPOCH value not found in 'Epoch' extensible codelist	Terminology	Warning
	EPOCH	FIRST SCREENING	CT2002	CG0021	EPOCH value not found in 'Epoch' extensible codelist	Terminology	Warning
	EPOCH	FIRST TREATMENT	CT2002	CG0021	EPOCH value not found in 'Epoch' extensible codelist	Terminology	Warning
	EPOCH	SECOND SCREENING	CT2002	CG0021	EPOCH value not found in 'Epoch' extensible codelist	Terminology	Warning
	EPOCH	SECOND TREATMENT	CT2002	CG0021	EPOCH value not found in 'Epoch' extensible codelist	Terminology	Warning
	Excess, Variable	10, STUDYID	SD1082	FDAB018	Variable length is too long for actual data	Metadata	Error
	Excess, Variable	10, USUBJID	SD1082	FDAB018	Variable length is too long for actual data	Metadata	Error
	Excess, Variable	184, EGMETHOD	SD1082	FDAB018	Variable length is too long for actual data	Metadata	Error
	Excess, Variable	190, EGORRES	SD1082	FDAB018	Variable length is too long for actual data	Metadata	Error
	Excess, Variable	190, EGSTRESC	SD1082	FDAB018	Variable length is too long for actual data	Metadata	Error
	Excess, Variable	194, EGPOS	SD1082	FDAB018	Variable length is too long for actual data	Metadata	Error
	Excess, Variable	20, VISIT	SD1082	FDAB018	Variable length is too long for actual data	Metadata	Error
	Excess, Variable	24, EPOCH	SD1082	FDAB018	Variable length is too long for actual data	Metadata	Error
	Excess, Variable	26, EGTEST	SD1082	FDAB018	Variable length is too long for actual data	Metadata	Error
	Excess, Variable	3, EGTESTCD	SD1082	FDAB018	Variable length is too long for actual data	Metadata	Error

log_rep=Y (default=N)

Records validation checks to the log

Useful for fully automated runs of datasets i.e. on receipt of a new data cut where log checkers would pick-up the data issues

```
%if &log_rep.=Y %then %do;  
  data _null_;  
    set p21_summary;  
    put "WARNING: P21 Check " P21_ID Pub_ID Message= Severity=  
  run;  
%end;
```

```
WARNING: P21 Check SD0009 FDAC206 MESSAGE=No qualifiers set to 'Y', when AE is Serious SEVERITY=Error COUNT=3  
WARNING: P21 Check SD0090 FDAC209 MESSAGE=AESDTH is not 'Y', when AEDOUT='FATAL' SEVERITY=Error COUNT=3  
WARNING: P21 Check SD0091 FDAC210 MESSAGE=AEDOUT is not 'FATAL', when AESDTH='Y' SEVERITY=Error COUNT=1  
WARNING: P21 Check SD1082 FDAC036 MESSAGE=Variable length is too long for actual data SEVERITY=Error COUNT=17  
WARNING: P21 Check SD0021 FDAC117 MESSAGE=Missing End Time-Point value SEVERITY=Warning COUNT=1173  
WARNING: P21 Check SD0031 FDAC122  
MESSAGE=Missing values for AESTDTC, AESTRF and AESTRTPT, when AEENDTC, AEENRF or AEENRTPT is provided SEVERITY=Warning COUNT=3  
WARNING: P21 Check SD1076 FDAC031 MESSAGE=Model permissible variable added into standard domain SEVERITY=Warning COUNT=1  
NOTE: There were 7 observations read from the data set WORK.P21_SUMMARY.  
NOTE: DATA statement used (Total process time):  
    real time           0.01 seconds  
    cpu time            0.01 seconds
```

Running the Macro

Intended to be called at the end of the SAS code used to produce a domain

Designed to be user-friendly and with a few simple parameters

```
*--- Validation of domain;
```

```
%p21_validate(study=TestStudy, model=SDTM, domain=EG);
```

```
*--- Validation of domain and full report validation issues  
(not just summary) displayed;
```

```
%p21_validate(study= TestStudy, model=SDTM, domain=EG,  
print_details=Y);
```

```
*--- Validation of ADaM ADSL, with warnings of issues printed  
to the log file;
```

```
%p21_validate(study= TestStudy, model=ADaM, domain=ADSL,  
log_rep=Y);
```

Macro Input Parameters

Parameter	Description
p21_ver	Version of P21 Community
study	Short name identifier for report file
model	Specify either SDTM/ADaM to be validated
model_version	Version of the CDISC model to be validated
terminology	Version of NCI Controlled Terminology
log_rep	Display results of validation to log for single domain validation (default: N)
print_summary	Print the summary of validation to SAS window for single domain validation (default: Y)
print_details	Print full details of validation to SAS window for single domain validation (default: N)
type=Excel	Validation report output file type (defaulted: Excel) (2.2.0 only)
cutoff=1000	Repeats of data validation issues in report (defaulted: 1000)
engine	FDA is 1903.1, PMDA is 1511.5 (3.0.1 only)
local	Select JAR from central location or run from local installation (3.0.1 only)

Utilising the Report

An understanding of what each of the P21 validation checks mean is critical for efficient use of the macro

Likely to be more open issues during development:

- Consider the completeness of the study
- Will we get more data?
- Data may still being entered into the database and resolve issues in time
- External data may not yet be received

Increased exposure for a developer to validation is beneficial rather than relying on just the lead study programmer to review the validation report

Encourages developers understand how to fix, spot and *avoid* issues at programming level earlier in development cycle

FDAC036 Check

Special note should be to mention that the following check will regularly fire:

FDAC036 - *“Variable length is too long for actual data”*

This is expected as only final datasets for submission should have their character variables trimmed down to the minimum required to store the complete variable and not necessarily our work version

PHASTAR have a utility macro when creating CRT packages to perform this task

Validation Automation

PHSATAR **%p21_validate** macro is also setup to validate all datasets together using the CLI method for complete crosscheck validation

PHASTAR also use a sister macro **%p21_define** to generates the define.xml from the P21 specification XLSX files as input

Thus enables automation within SAS to harness P21 Community for a large part of the CRT validation process to:

- create the define.xml
- create and compress XPT
- validate the define.xml/XPT

Conclusion

By adopting this macro the programming team at PHASTAR are able to:

- Identify and correct potential dataset specification issues
- Identify raw data issues that have impacted SDTM
- Identify SDTM issues before they used for ADaM
- Identify compliance issues before double-programming

Correcting issues earlier in the developmental process of creates a more efficient process as there is less likely the need to retrospectively make a “*small fix*” to a dataset for CDISC compliance

Further Discussion

Macro code is future-proofed by having distinct sections for each release of P21 so it can be up-versioned and still be backwards compatible

Update from P21 2.2.0 to 3.0.1 was a substantial change, mostly driven by the way P21 Community now installs in a local user desktop rather than centrally which affects the macro (required extra macro input param)

A similar process could be adapted for working with P21 Enterprise – see P21 paper referenced in associated paper

PHASTAR control default parameters into the macro using a global macro to set these values centrally. This saves on the number of input parameters, or it can be adjusted at a study level for make static versioning across the study

Any Questions?

David Meek

PHASTAR

Unit 2, 2a Bollo Lane

Chiswick, London, W4 5LE

Email: david.meek@PHASTAR.com

Web: www.PHASTAR.com