PharmaSUG 2014 – Paper HT07

SDTM, ADaM and define.xml with OpenCDISC®

Angela Ringelberg, Ockham Onocology, Cary, NC
Tracy Sherman, inVentiv Health Clinical, Cary, NC

ABSTRACT

As programmers, many of us have spent hours reviewing SDTM/ADaM standards and implementation guides to generate "compliant" CDISC SAS data sets. However, there is an easier way to ensure compliance with CDISC standards, including SDTM, ADaM, Define.xml, and others.

OpenCDISC® is an open source community which is focusing on creating frameworks and tools for the implementation and advancement of CDISC Standards. OpenCDISC® has created a CDISC Validator which will eliminate the need for individuals to develop their own custom processes in order to ensure that their CDISC models are compliant with CDISC standards. By taking common validation rules, OpenCDISC® has developed an open-source tool which is freely available and of commercial-quality to ensure data compliance with CDISC models such as SDTM, ADaM and Define.xml. The validation rules for each standard have been pooled into a CDISC Validation Rules Repository, providing users with a central listing. The listing is easy to use, modify and adapt.

In this Hands-On Training, we are going to briefly describe a few of the key terms (SDTM, ADaM, Define.xml) and investigate the use of OpenCDISC Validator to perform the validation of SDTM, ADaM and define.xml.

INTRODUCTION

Before discussing the details of OpenCDISC, a quick overview of CDISC (Clinical Data Interchange Standards Consortium) is merited. Over the past few years CDISC has become common terminology in our workplace and we have started to use CDISC standards in our work more and more. The CDISC standards provide data consistency across the spectrum and this standardization has helped streamline drug development.

In this paper, we are going to concentrate on the SDTM, ADaM, and define.xml CDISC standards. SDTM (Study Data Tabulation Model) is the content standard of case report form data tabulations from clinical research studies. ADaM (Analysis Data Model) is the content standard of analysis datasets. Define.xml (Case Report Tabulation Data Definition Specification (CRTDDS)) is an XML-based content and format standard which contains the specifications for data definitions for CDISC SDTM datasets.

When we create SDTM files, ADaM files, and/or define.xml, we must make sure that they are compliant with CDISC standards. We must check our work. How is this done? The task is usually done by double programming (at least in the case of SDTM and ADaM files); the re-creation of the files by an independent programmer and comparing the two sets of results. This is no simple task. It requires a lot of time and a lot of reconciliation between the production programming and the validation programming in order to make sure there is compliance with CDISC standards. And, once this process is complete, how can we guarantee 100% compliance? The individualized validation process for compliance with the CDISC standards is not a standardized task; each of us develops our own ways of validating our files.

Here is where OpenCDISC comes into the picture. OpenCDISC has created a CDISC Validator which will eliminate the need for individuals to develop their own custom processes. The OpenCDISC Validator ensures that your CDISC models are compliant with CDISC standards. OpenCDISC has taken common validation rules and pooled them into a CDISC Validation Rules Repository providing users with a central listing. The Validator is free and easy to use.

USING THE OPENCDISC VALIDATOR

The validator requires Java Runtime Environment (JRE) version 1.6 or higher and 2GB system RAM. Download the OpenCDISC validator from http://www.opencdisc.org, click on the OpenCDISC Validator link [select v1.4.1 for the most recent release or select a previous release] and unzip to your chosen directory. Detailed installation directions are provided on the website.

Figure 1. The OpenCDISC download page

Once the validator has been downloaded and unzipped, it is ready to use.

VALIDATING SDTM FILES

Step 1: Open the 'opendisc-validator' folder.

Step 2: Double click on the 'client.bat' file. This will bring up the OpenCDISC Validator window:

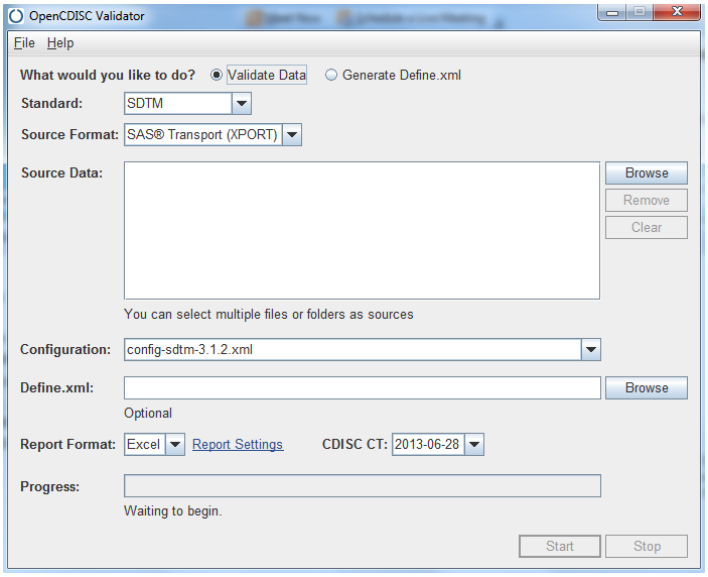


Figure 2. The OpenCDISC interface

Step 3: For the question "What would you like to do?' select 'Validate Data'.

Step 4: Choose the Standard (the default is SDTM). For this example, we chose SDTM.

Step 5: Choose the Format (the default is XPORT). Note that the SDTM files must be in SAS® Transport (XPORT) or a delimited file. If you select 'delimited', then you can specify a delimiter. The default delimeter is a '|' (vertical pipe). The validator cannot process regular SAS datasets.

Step 6: Choose the source data by clicking on the Browse button on the right hand side. The following window will appear once you change the directory to a location that contains SAS XPT files:

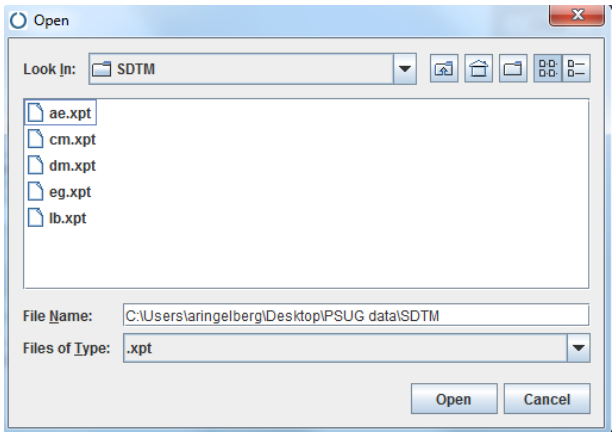


Figure 3. The dataset-selection dialog

Step 7: Highlight the SDTM file or files you want to check. Note that the file(s) you have selected will be listed in the File Name location.

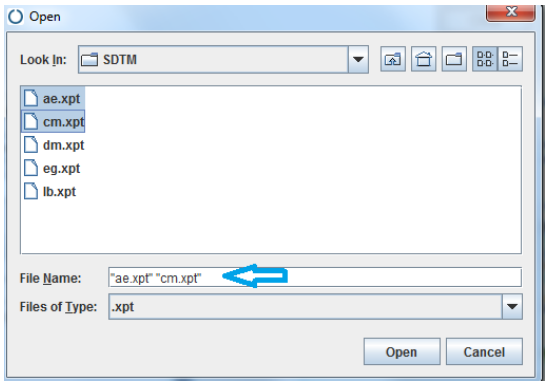


Figure 4. The dataset-selection dialog during file selection

Step 8: Click Open. The OpenCDISC validator window will appear with the file(s) you have selected in the Source Data field.

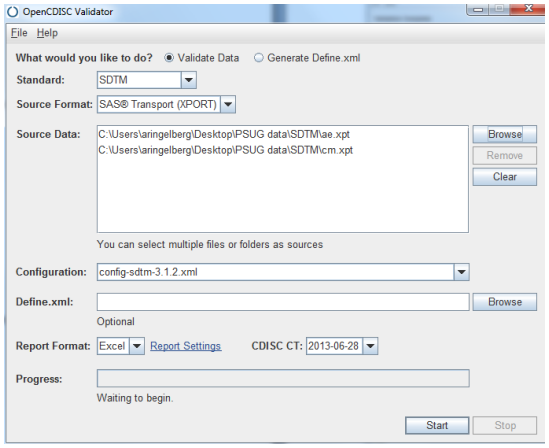


Figure 5. The dataset-selection dialog after file selection

Step 9: Choose the configuration. The default for SDTM files is sdtm-3.1.2.xml.

Step 10: Choose the report format. The default is Excel.

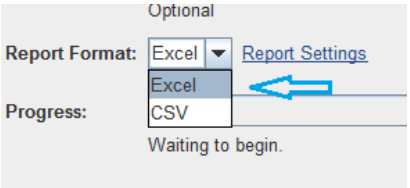


Figure 6. The Report Format selection

Step 11: Choose the CDISC CT (Controled Terminology) version you want to work with. The default is the most recent version.

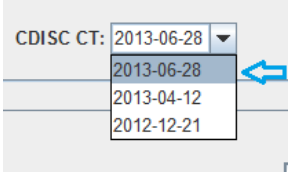


Figure 7. The Controlled Terminology selection

The 1.4 version of the validator currently ships with SDTM terminologies of 2012-12-21, 2012-08-03 and 2012-06-29. However, since CDISC Controlled terminology has variable release dates through out the year, there may be a need to update the Controlled terminology. Downloading the most recent version is simple. Refer to the OpenCDISC instructions at http://www.opencdisc.org/configuring-opencdisc-validator-cdisc-controlled-terminology under the 'Adding CDISC CT versions for validation' for complete instructions.

Step 12: In this example, we are now ready to start the validation of the AE and CM SDTM files. Click the start button. Note that the number of datasets that have been processed is captured in the Progress status bar:

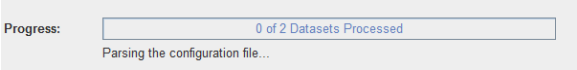


Figure 8. The Progress window

When the validation is complete, you will receive an information window providing you with how long it took for the validator to run, the number of records examined, the number of datasets validated, the number of messages generated and the number of checks performed:

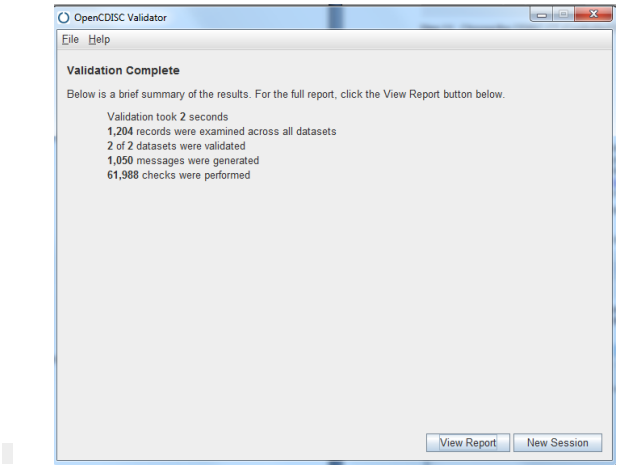
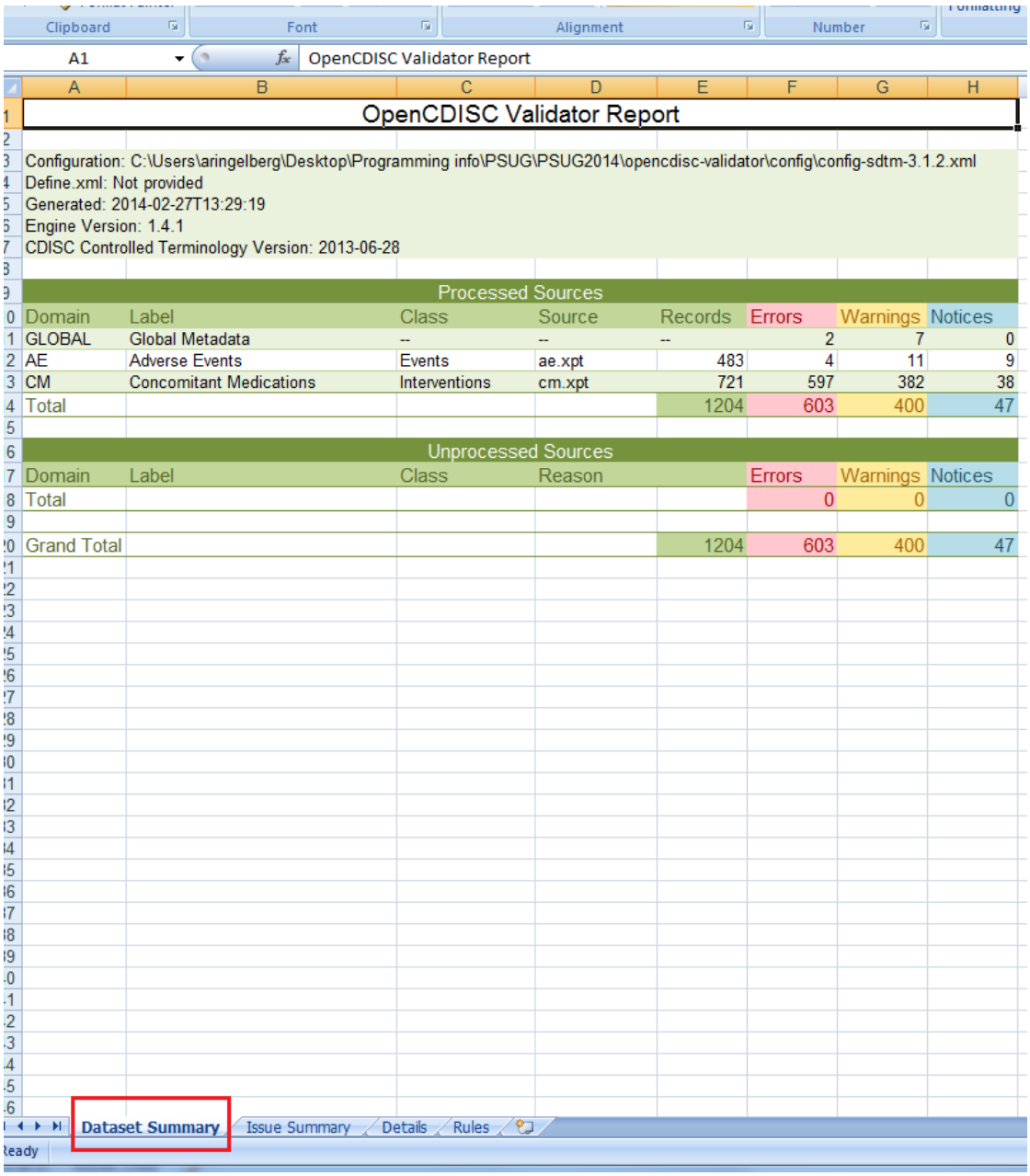


Figure 9. The validation summary report

Step 13: At this point, you may choose to view the report or start a new session. Choose 'View Report'. The report will consist of 4 tabs within the Excel document: Dataset Summary, Issue Summary, Details and Rules.

The **Dataset Summary** tab provides a brief overview of what was encountered by the validator.



OpenCDISC Validator Report

Configuration: C:\Users\aringelberg\Desktop\Programming info\PSUG\PSUG2014\opencdisc-validator\config\config-sdtm-3.1.2.xml
Define.xml: Not provided
Generated: 2014-02-27T13:29:19
Engine Version: 1.4.1
CDISC Controlled Terminology Version: 2013-06-28

Processed Sources

Domain	Label	Class	Source	Records	Errors	Warnings	Notices
GLOBAL	Global Metadata	--	--	--	2	7	0
AE	Adverse Events	Events	ae.xpt	483	4	11	9
CM	Concomitant Medications	Interventions	cm.xpt	721	597	382	38
Total				1204	603	400	47

Unprocessed Sources

Domain	Label	Class	Reason	Errors	Warnings	Notices
Total				0	0	0

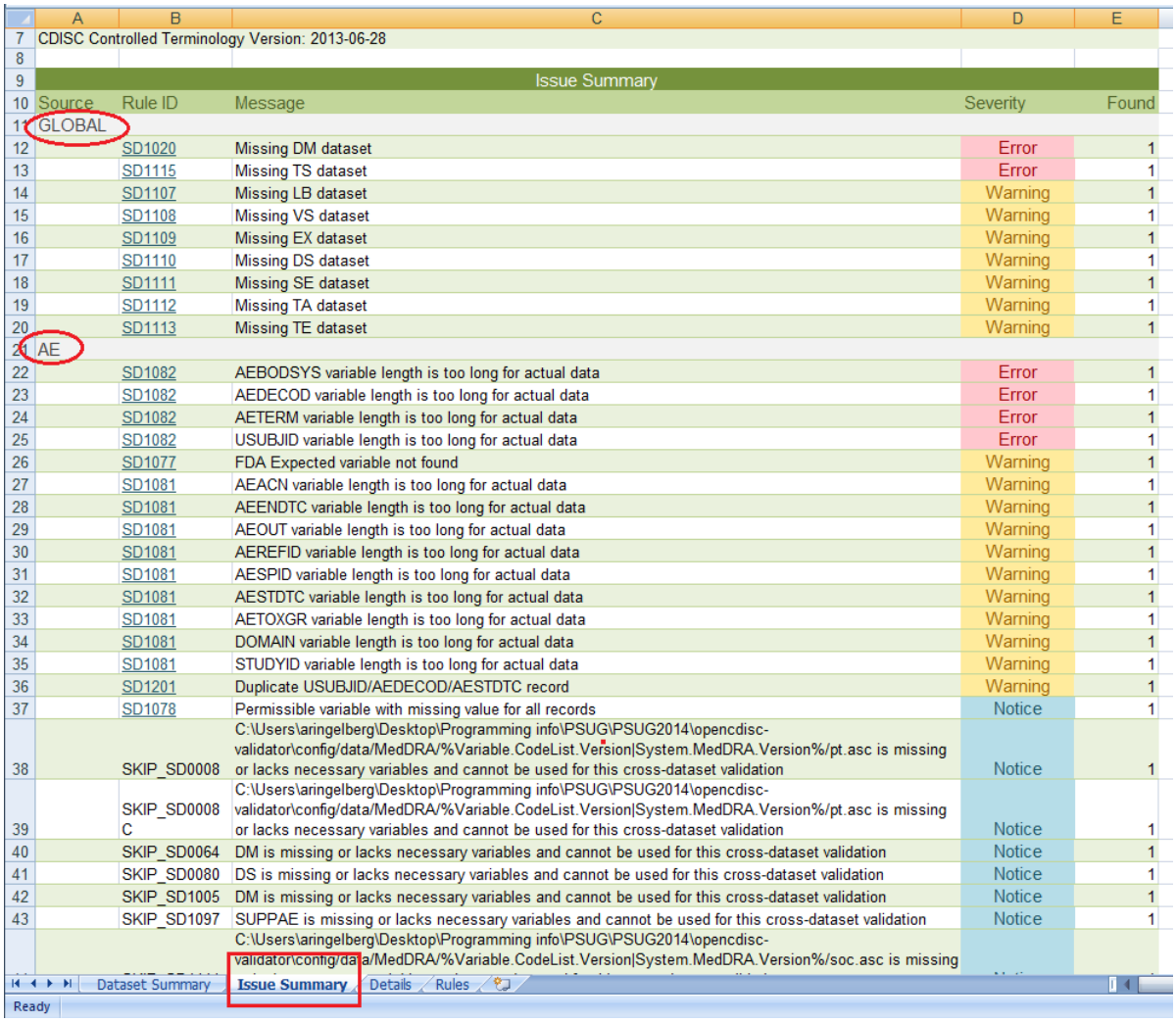
Grand Total				1204	603	400	47

Dataset Summary | Issue Summary | Details | Rules

Figure 10. The Dataset Summary window

The report highlights what the validation encountered. In this example, it is reporting issues found in the two files, AE and CM. These are the only two files we requested to be validated. In the Dataset Summary tab, we can see the number of records, errors, warnings and notices that were generated for each of the files. For example, the AE file contains 4 errors, 11 warnings and 9 notices. Looking at the CM file, we can see the number of errors is much higher with a report of 597 errors, 382 warnings and 38 notices. It can be bit daunting to see so many errors being reported, but perhaps there is a general issue with the file that is being duplicated across multiple observations. To investigate this further, we can look on the **Issue Summary** tab.

The **Issues Summary** tab provides a break down of the type of rules that have issues and how many have been reported by source. In the screenshot below the GLOBAL and AE errors, warnings and notices can be seen. And if you were able to scroll down, you would also see the DM issues. The Issues Summary tab provides a bit more detail as to what is being reported with each file.

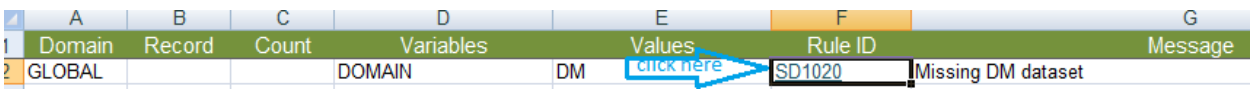


	A	B	C	D	E
7	CDISC Controlled Terminology Version: 2013-06-28				
8					
9			Issue Summary		
10	Source	Rule ID	Message	Severity	Found
11	GLOBAL				
12		SD1020	Missing DM dataset	Error	1
13		SD1115	Missing TS dataset	Error	1
14		SD1107	Missing LB dataset	Warning	1
15		SD1108	Missing VS dataset	Warning	1
16		SD1109	Missing EX dataset	Warning	1
17		SD1110	Missing DS dataset	Warning	1
18		SD1111	Missing SE dataset	Warning	1
19		SD1112	Missing TA dataset	Warning	1
20		SD1113	Missing TE dataset	Warning	1
21	AE				
22		SD1082	AEBODSYS variable length is too long for actual data	Error	1
23		SD1082	AEDECOD variable length is too long for actual data	Error	1
24		SD1082	AETERM variable length is too long for actual data	Error	1
25		SD1082	USUBJID variable length is too long for actual data	Error	1
26		SD1077	FDA Expected variable not found	Warning	1
27		SD1081	AEACN variable length is too long for actual data	Warning	1
28		SD1081	AEENDTC variable length is too long for actual data	Warning	1
29		SD1081	AEOUT variable length is too long for actual data	Warning	1
30		SD1081	AEREFID variable length is too long for actual data	Warning	1
31		SD1081	AESPID variable length is too long for actual data	Warning	1
32		SD1081	AESTDTC variable length is too long for actual data	Warning	1
33		SD1081	AETOXGR variable length is too long for actual data	Warning	1
34		SD1081	DOMAIN variable length is too long for actual data	Warning	1
35		SD1081	STUDYID variable length is too long for actual data	Warning	1
36		SD1201	Duplicate USUBJID/AEDECOD/AESTDTC record	Warning	1
37		SD1078	Permissible variable with missing value for all records	Notice	1
38		SKIP_SD0008	C:\Users\aringelberg\Desktop\Programming info\PSUG\PSUG2014\opencdisc-validator\config/data/MedDRA/%Variable.CodeList.Version|System.MedDRA.Version%/pt.asc is missing or lacks necessary variables and cannot be used for this cross-dataset validation	Notice	1
39		SKIP_SD0008C	C:\Users\aringelberg\Desktop\Programming info\PSUG\PSUG2014\opencdisc-validator\config/data/MedDRA/%Variable.CodeList.Version|System.MedDRA.Version%/pt.asc is missing or lacks necessary variables and cannot be used for this cross-dataset validation	Notice	1
40		SKIP_SD0064	DM is missing or lacks necessary variables and cannot be used for this cross-dataset validation	Notice	1
41		SKIP_SD0080	DS is missing or lacks necessary variables and cannot be used for this cross-dataset validation	Notice	1
42		SKIP_SD1005	DM is missing or lacks necessary variables and cannot be used for this cross-dataset validation	Notice	1
43		SKIP_SD1097	SUPPAE is missing or lacks necessary variables and cannot be used for this cross-dataset validation	Notice	1
			C:\Users\aringelberg\Desktop\Programming info\PSUG\PSUG2014\opencdisc-validator\config/data/MedDRA/%Variable.CodeList.Version|System.MedDRA.Version%/soc.asc is missing		

Figure 11. The Issue Summary window

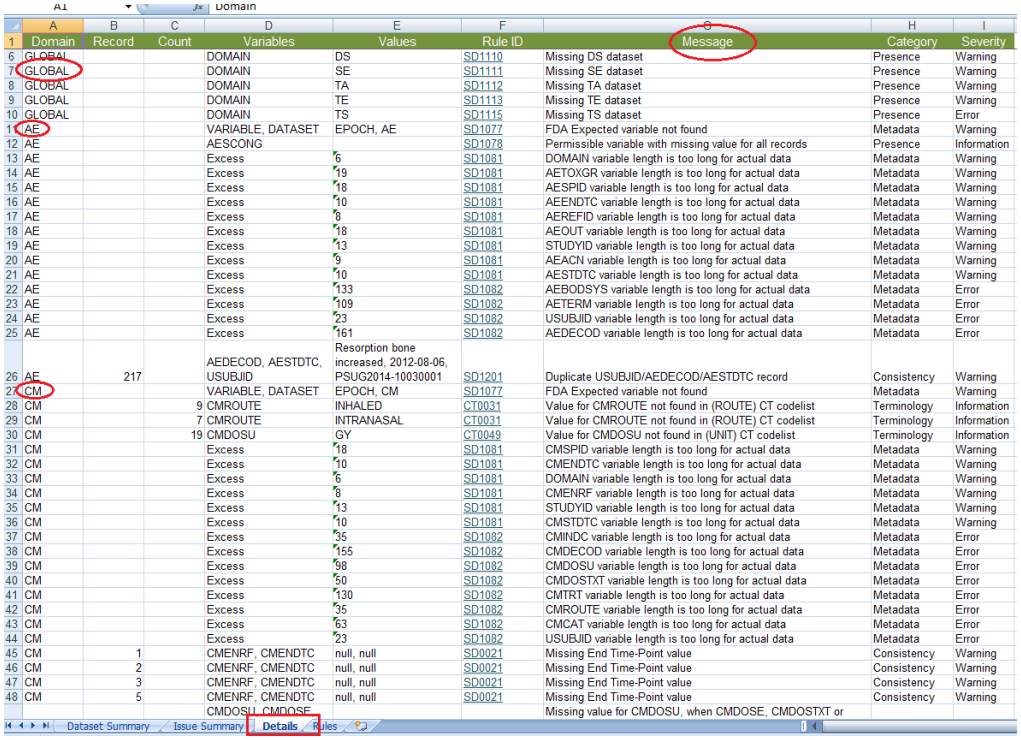
The **Details** tab provides us just that…the details. Each error or warning message has been expanded. In the second screen shot below, you can see which domain is affected and within that domain which variables, values, the specific rule ID, the message, the category and severity of the message.

If the Details tab does not provide enough insight on the error/warning/notice being generated, you can click on the Rules tab to see a list of all rules or just click on the Rules ID link on the Details tab for the specific message that needs to be investigated.



	A	B	C	D	E	F	G
1	Domain	Record	Count	Variables	Values	Rule ID	Message
2	GLOBAL			DOMAIN	DM	SD1020	Missing DM dataset

Figure 12. The Details Tab Rule ID



Domain	Record	Count	Variables	Values	Rule ID	Message	Category	Severity
GLOBAL			DOMAIN	DS	SD1110	Missing DS dataset	Presence	Warning
GLOBAL			DOMAIN	SE	SD1111	Missing SE dataset	Presence	Warning
GLOBAL			DOMAIN	TA	SD1112	Missing TA dataset	Presence	Warning
GLOBAL			DOMAIN	TE	SD1113	Missing TE dataset	Presence	Warning
GLOBAL			DOMAIN	TS	SD1115	Missing TS dataset	Presence	Error
AE			VARIABLE, DATASET	EPOCH, AE	SD1077	FDA Expected variable not found	Metadata	Warning
AE			AESCONG		SD1078	Permissible variable with missing value for all records	Presence	Information
AE			Excess	6	SD1081	DOMAIN variable length is too long for actual data	Metadata	Warning
AE			Excess	19	SD1081	AETOXGR variable length is too long for actual data	Metadata	Warning
AE			Excess	18	SD1081	AESPID variable length is too long for actual data	Metadata	Warning
AE			Excess	10	SD1081	AEENDTC variable length is too long for actual data	Metadata	Warning
AE			Excess	8	SD1081	AEREFID variable length is too long for actual data	Metadata	Warning
AE			Excess	18	SD1081	AEOUT variable length is too long for actual data	Metadata	Warning
AE			Excess	13	SD1081	STUDYID variable length is too long for actual data	Metadata	Warning
AE			Excess	9	SD1081	AEACN variable length is too long for actual data	Metadata	Warning
AE			Excess	10	SD1081	AESTDTC variable length is too long for actual data	Metadata	Warning
AE			Excess	133	SD1082	AEBODSYS variable length is too long for actual data	Metadata	Error
AE			Excess	109	SD1082	AETERM variable length is too long for actual data	Metadata	Error
AE			Excess	23	SD1082	USUBJID variable length is too long for actual data	Metadata	Error
AE			Excess	161	SD1082	AEDECOD variable length is too long for actual data	Metadata	Error
AE	217		AEDECOD, AESTDTC, USUBJID	Resorption bone increased, 2012-08-06, PSUG2014-10030001	SD1201	Duplicate USUBJID/AEDECOD/AESTDTC record	Consistency	Warning
CM			VARIABLE, DATASET	EPOCH, CM	SD1077	FDA Expected variable not found	Metadata	Warning
CM		9	CMROUTE	INHALED	CT0031	Value for CMROUTE not found in (ROUTE) CT codelist	Terminology	Information
CM		7	CMROUTE	INTRANASAL	CT0031	Value for CMROUTE not found in (ROUTE) CT codelist	Terminology	Information
CM		19	CMDOSU	GY	CT0049	Value for CMDOSU not found in (UNIT) CT codelist	Terminology	Information
CM			Excess	18	SD1081	CMSPID variable length is too long for actual data	Metadata	Warning
CM			Excess	10	SD1081	CMENDTC variable length is too long for actual data	Metadata	Warning
CM			Excess	6	SD1081	DOMAIN variable length is too long for actual data	Metadata	Warning
CM			Excess	8	SD1081	CMENRF variable length is too long for actual data	Metadata	Warning
CM			Excess	13	SD1081	STUDYID variable length is too long for actual data	Metadata	Warning
CM			Excess	10	SD1081	CMSTDTC variable length is too long for actual data	Metadata	Warning
CM			Excess	35	SD1082	CMINDC variable length is too long for actual data	Metadata	Error
CM			Excess	155	SD1082	CMDECOD variable length is too long for actual data	Metadata	Error
CM			Excess	98	SD1082	CMDOSU variable length is too long for actual data	Metadata	Error
CM			Excess	50	SD1082	CMDOSTXT variable length is too long for actual data	Metadata	Error
CM			Excess	130	SD1082	CMTRT variable length is too long for actual data	Metadata	Error
CM			Excess	35	SD1082	CMROUTE variable length is too long for actual data	Metadata	Error
CM			Excess	63	SD1082	CMCAT variable length is too long for actual data	Metadata	Error
CM			Excess	23	SD1082	USUBJID variable length is too long for actual data	Metadata	Error
CM	1		CMENRF, CMENDTC	null, null	SD0021	Missing End Time-Point value	Consistency	Warning
CM	2		CMENRF, CMENDTC	null, null	SD0021	Missing End Time-Point value	Consistency	Warning
CM	3		CMENRF, CMENDTC	null, null	SD0021	Missing End Time-Point value	Consistency	Warning
CM	5		CMENRF, CMENDTC	null, null	SD0021	Missing End Time-Point value	Consistency	Warning
			CMDOSU, CMDOSE			Missing value for CMDOSU, when CMDOSE, CMDOSTXT or		

Dataset Summary | Issue Summary | Details | Rules

Figure 13. The Details window

And finally, the **Rules** tab shows us the standard checks implemented by the validator. The error report points to those rules that we need to review and learn about. It narrows down the large volume of SDTM concepts and kind of acts like a self-directed CDISC 'training'. The screen shot below shows a small sample of some of the rules with their descriptions.

Rule ID	Message	Description	Category	Severity
SD0082	Exposure end date is after the latest Disposition date	End Date/Time of Treatment (EXENDTC) should be less than or equal to the Start Date/Time of the latest Disposition Event (DSSTDTC)	Consistency	Warning
SD0083	Duplicate USUBJID	The value of Unique Subject Identifier (USUBJID) variable must be unique for each subject across all trials in the submission	Consistency	Error
SD0084	Negative value for AGE	The value of Age (AGE) cannot be less than 0	Limit	Error
SD0085	Mismatch between IEORRES and IESTRESC values	I/E Criterion Original Result (IEORRES) and I/E Criterion Result in Std Format (IESTRESC) should have the same value	Consistency	Warning
SD0086	SUPPQUAL duplicate records	All SUPPQUAL Domains records must have unique combination of Study Identifier (STUDYID), Unique Subject Identifier (USUBJID), Identifying Variable (IDVAR), Identifying Variable Value (IDVARVAL) and Qualifier Variable Name (QNAM) variables values	Consistency	Error
SD0087	RFSTDTC is not provided for a randomized subject	Subject Reference Start Date/Time (RFSTDTC) should be populated for all randomized subjects, those where Planned Arm Code (ARMCD) is not equal to 'SCRNFAIL' or 'NOTASSGN'	Consistency	Warning
SD0088	RFENDTC is not provided for a randomized subject	Subject Reference End Date/Time (RFENDTC) should be populated for all randomized subjects, those where Planned Arm Code (ARMCD) is not equal to 'SCRNFAIL' or 'NOTASSGN'	Consistency	Warning
SD0089	Missing values for TEENRL and TEDUR	At least one of Rule for End of Element (TEENRL) or Planned Duration of Element (TEDUR) should be populated	Consistency	Warning
SD0090	AESDTH is not 'Y', when AEOUT='FATAL'	Results in Death (AEDTH) should equal 'Y', when Outcome of Adverse Event (AEOUT) is 'FATAL'	Consistency	Warning
SD0091	AEOUT is not 'FATAL', when AESDTH='Y'	Outcome of Adverse Event should equal 'FATAL', when Results in Death (AEDTH) is 'Y'	Consistency	Warning
SD0092	Missing value for SEUPDES, when ETCD='UNPLAN'	Description of Unplanned Element (SEUPDES) should be populated, when subject's experience for a particular period of time is represented as an unplanned Element, where Element Code (ETCD) is equal to 'UNPLAN'	Consistency	Warning
SD0093	Missing value for AGEU, when AGE is provided	Age Units (AGEU) should be provided, when Age (AGE) is populated	Consistency	Error
SD0094	DSCAT is not 'DISPOSITION EVENT', when EPOCH is provided	Category for Disposition Event (DSCAT) should equal 'DISPOSITION EVENT', when Epoch (EPOCH) is provided	Consistency	Warning
SD0095	SUPPQUAL dataset is used for non-general-observation-class Domain	Supplemental Qualifiers special purpose dataset (SUPP--) can only be used to capture non-standard variables and their association to parent records in general-observation-class datasets (Events, Findings, Interventions) and Demographics	Presence	Error
SD1001	Duplicate SUBJID	The value of Subject Identifier for the Study (SUBJID) variable must be unique for each subject within the study	Consistency	Error
SD1002	RFSTDTC is after RFENDTC	Subject Reference Start Date/Time (RFSTDTC) must be less than or equal to Subject Reference End Date/Time (RFENDTC)	Limit	Error
SD1003	Missing value for AGE, when AGEU is provided	Age (AGE) should be provided, when Age Units (AGEU) are populated	Consistency	Error
SD1004	Invalid value for ARMCD	The value of Planned Arm Code (ARMCD) should be no more than 20 characters in length	Format	Warning
SD1005	Invalid STUDYID	Study Identifier (STUDYID) values must match the STUDYID in Demographics (DM) domain	Consistency	Error
SD1008	CODTC is populated, when comment is a child record of another domain	The value of Date/Time of Comment (CODTC) should be NULL, when comments are related to a specific parent record or group of parent records in a domain (RDOMAIN, IDVAR and IDVARVAL are populated)	Consistency	Warning
SD1009	Invalid value for ETCD	The value of Element Code (ETCD) should be no more than 8 characters in length	Format	Warning
SD1010	ELEMENT value is populated, when ETCD='UNPLAN'	Description of Element (ELEMENT) should be NULL, when subject's experience for a particular period of time is represented as an unplanned Element, where Element Code (ETCD) is equal to 'UNPLAN'	Consistency	Warning
SD1011	Invalid ISO 8601 value for -- variable	Value of Duration, Elapsed Time, and Interval variables (--DUR, --ELTM, --EVLINT) must conform to the ISO 8601 international standard	Format	Error

Dataset Summary | Issue Summary | Details | Rules

Figure 14. The Rules window

With the information obtained from the OpenCDISC validator, the user can now go back to the production SDTM file and correct any issues the validator has flagged.

VALIDATING ADAM FILES

The steps needed for validating ADaM files are very similar to those needed for SDTM validation:

Step 1: Open the 'opendisc-validator' folder.

Step 2: Double click on the 'client.bat' file. This will bring up the OpenCDISC Validator window.

Step 3: For the question 'What would you like to do?' select 'Validate Data'.

Step 4: Choose the Standard (the default is SDTM). For this example, we chose **ADaM**.

Step 5: Choose the Format (the default is XPORT).

Step 6: Choose the source data by clicking on the Browse button.

Step 7: Highlight the ADaM file or files you want to check. Please note that the validator requires the DM and TS SDTM file to do cross checks so it's a good idea to include the DM and TS xpt files at this point.

Step 8: Click Open. The OpenCDISC validator window will appear with the files or files you have selected in the Source Data field.

Step 9: Choose the configuration. Select 'config-adam-1.0.xml'. This is the default for ADaM files, and currently your only choice.

Step 10: Choose the report format. The default is Excel.

Step 11: Choose the CDISC CT (Controlled terminology) version. The default is the most recent version that was available at the release time of the validator. If newer CDISC CT is needed, you will need to download that version. See step 11 in the SDTM explanation above for instructions on how to download newer CDISC CT versions.

Step 12: Once you have selected (steps 6 -8) which file(s) needs validation, click the **Start** button.

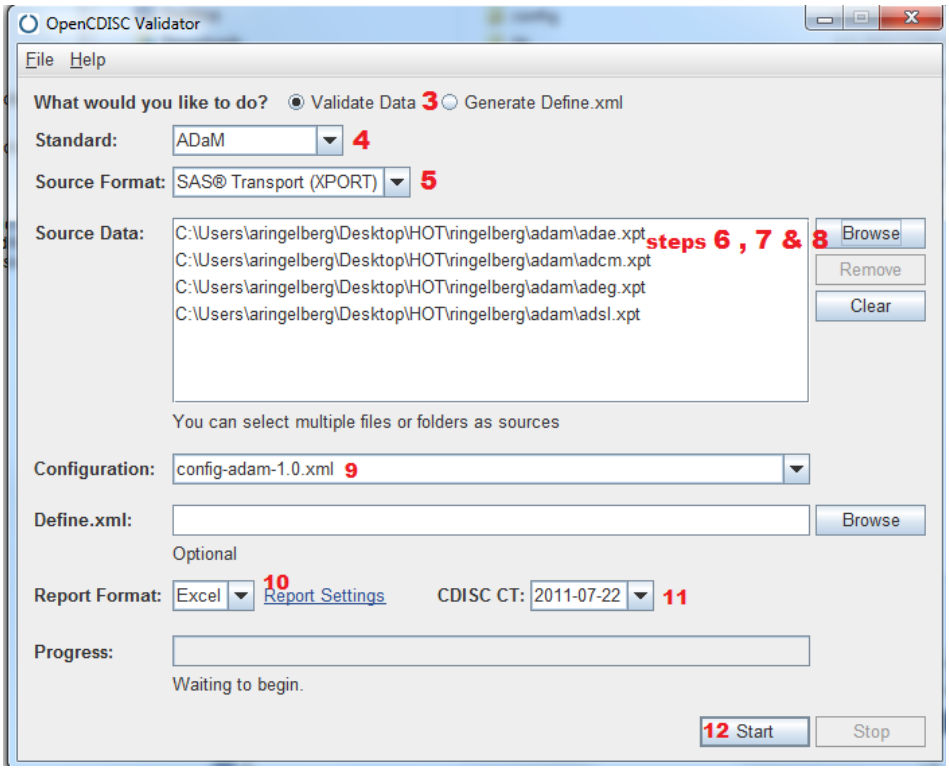


Figure 15. The Open CDISC interface choosing ADaM validation

When the validation is complete, you will receive the information window providing you the same type of summary information as when we ran the STDM file validation.

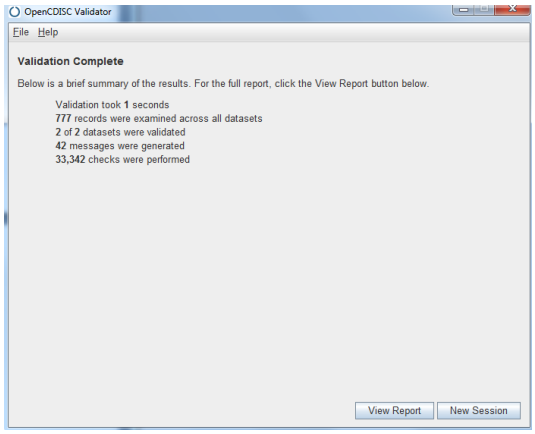


Figure 16. The summary report generated for the ADaM validation

Step 12: At this point, you may choose to view the report or start a new session. Choose 'View Report'. As with the SDTM validation, the ADaM validation provides a similar OpenCDISC Validator Report containing 4 tabs: Dataset Summary, Issue Summary, Details and Rules.

OpenCDISC Validator Report

Configuration: C:\Users\aringelberg\Desktop\HOT\ringelberg\OpenCDISC v1.4.1\opencdisc-validator\config\config-adam-1.0.xml
Define.xml: Not provided
Generated: 2014-04-01T12:17:35
Engine Version: 1.4.1
CDISC Controlled Terminology Version: 2011-07-22

Processed Sources

Domain	Label	Class	Source	Records	Errors	Warnings	Notices
GLOBAL	Global Metadata	--	--	--	0	0	0
ADEG	Basic Data Structure	Basic Data Structure	adeg.xpt	695	31	0	1
ADSL	Subject-Level Analysis	Subject-Level Analysis	adsl.xpt	82	0	0	8
Total				777	31	0	9

Unprocessed Sources

Domain	Label	Class	Reason	Errors	Warnings	Notices
ADAE	Unknown	Unknown	Configuration Missing	0	1	0
ADCM	Unknown	Unknown	Configuration Missing	0	1	0
Total				0	2	0
Grand Total				777	31	2	9

Figure 17. The Dataset Summary for the ADaM validation

OpenCDISC Validator Report

Configuration: C:\Users\aringelberg\Desktop\HOT\ringelberg\OpenCDISC v1.4.1\opencdisc-validator\config\config-adam-1.0.xml
Define.xml: Not provided
Generated: 2014-04-01T12:17:35
Engine Version: 1.4.1
CDISC Controlled Terminology Version: 2011-07-22

Issue Summary

Source	Rule ID	Message	Severity	Found
ADAE				
	MISSING_CONFIG	Unrecognized domain	Warning	1
ADCM				
	MISSING_CONFIG	Unrecognized domain	Warning	1
ADEG				
	AD0018	Variable label mismatch between dataset and ADaM standard	Error	1
	AD0177	Multiple baseline records exist for a unique USUBJID,PARAMCD,BASETYPE	Error	30
	SKIP_AD0053	DM is missing or lacks necessary variables and cannot be used for this cross-dataset validation	Notice	1
ADSL				
	SKIP_AD0053	DM is missing or lacks necessary variables and cannot be used for this cross-dataset validation	Notice	1
	SKIP_AD0204	DM is missing or lacks necessary variables and cannot be used for this cross-dataset validation	Notice	1
	SKIP_AD0205	DM is missing or lacks necessary variables and cannot be used for this cross-dataset validation	Notice	1
	SKIP_AD0206	DM is missing or lacks necessary variables and cannot be used for this cross-dataset validation	Notice	1
	SKIP_AD0207	DM is missing or lacks necessary variables and cannot be used for this cross-dataset validation	Notice	1
	SKIP_AD0208	DM is missing or lacks necessary variables and cannot be used for this cross-dataset validation	Notice	1
	SKIP_AD0209	DM is missing or lacks necessary variables and cannot be used for this cross-dataset validation	Notice	1
	SKIP_AD0210	DM is missing or lacks necessary variables and cannot be used for this cross-dataset validation	Notice	1

Figure 18. The Issues Summary for the ADaM validation

Domain	Record	Count	Variables	Values	Rule ID	Message	Category	Severity
ADAE					MISSING_CONFIG	Unrecognized domain	System	Warning
ADCM					MISSING_CONFIG	Unrecognized domain	System	Warning
ADEG			VARIABLE, LABEL	AVISIT, Analysis Visit Name	AD0018	Variable label mismatch between dataset and ADaM standard	Metadata	Error
ADEG	88		ABLFL, PARAMCD, USUBJID	Y, HRMEAN, PSUG2014-10010001	AD0177	Multiple baseline records exist for a unique USUBJID,PARAMCD,BASETYPE	Consistency	Error
ADEG	100		ABLFL, PARAMCD, USUBJID	Y, INTP, PSUG2014-10010001	AD0177	Multiple baseline records exist for a unique USUBJID,PARAMCD,BASETYPE	Consistency	Error
ADEG	112		ABLFL, PARAMCD, USUBJID	Y, PRMEAN, PSUG2014-10010001	AD0177	Multiple baseline records exist for a unique USUBJID,PARAMCD,BASETYPE	Consistency	Error
ADEG	124		ABLFL, PARAMCD, USUBJID	Y, QRSDUR, PSUG2014-10010001	AD0177	Multiple baseline records exist for a unique USUBJID,PARAMCD,BASETYPE	Consistency	Error
ADEG	136		ABLFL, PARAMCD, USUBJID	Y, QTMEAN, PSUG2014-10010001	AD0177	Multiple baseline records exist for a unique USUBJID,PARAMCD,BASETYPE	Consistency	Error

Figure 19. The Details tab for the ADaM validation

INTERPRETING THE OPENCDISC VALIDATOR OUTPUT

As we have seen, for both SDTM and ADaM files, the OpenCDISC Validator generates the same type of report. In the examples above, we requested that our report be put into Excel format. Within the Excel spreadsheet, the validator generated four information tabs, providing different levels of information on each tab. However, prior to viewing the report, the validator provides some feedback as to what it has encountered. This synopsis of the validation report gives the number of records that were read, the number of messages that were generated, and the number of checks that were performed. The number of messages generated can be a bit overwhelming and is really not very helpful in telling you what issues you have in your files. So, you need to take a closer look by viewing each of the report tabs generated by the validator.

The Dataset Summary tab gives an overview of the issues encountered in the validation. Looking at the CM file, we see that 597 error and 585 warning messages were generated. That is a lot of errors and warnings, let's take a closer look.

OpenCDISC Validator Report

Configuration: C:\Users\aringelberg\Desktop\HOT\ringelberg\OpenCDISC v1.4.1\opencdisc-validator\config\config-sdtm-3.1.2.xml
Define.xml: Not provided
Generated: 2014-04-01T13:16:40
Engine Version: 1.4.1
CDISC Controlled Terminology Version: 2013-06-28

Processed Sources

Domain	Label	Class	Source	Records	Errors	Warnings	Notices
GLOBAL	Global Metadata	--	--	--	1	8	0
CM	Concomitant Medications	Interventions	cm.xpt	721	597	585	35
DM	Demographics	Special Purpose Domains	dm.xpt	82	3	52	11
Total				803	601	645	46

Unprocessed Sources

Domain	Label	Class	Reason		Errors	Warnings	Notices
Total					0	0	0
Grand Total				803	601	645	46

Figure 20. The Dataset Summary - interpreting OpenCDISC

The Issue Summary tab provides a bit more detail. Breaking down the Error messages and Warning messages by type.

	SD1112	Missing TA dataset	Warning	1
	SD1113	Missing TE dataset	Warning	1
CM				
	SD0035	Missing value for CMDOSU, when CMDOSE, CMDOSTXT or CMDOSTOT is provided	Error	589
	SD1082	CMCAT variable length is too long for actual data	Error	1
	SD1082	CMDECOD variable length is too long for actual data	Error	1
	SD1082	CMDOSTXT variable length is too long for actual data	Error	1
	SD1082	CMDOSU variable length is too long for actual data	Error	1
	SD1082	CMINDC variable length is too long for actual data	Error	1
	SD1082	CMROUTE variable length is too long for actual data	Error	1
	SD1082	CMTRT variable length is too long for actual data	Error	1
	SD1082	USUBJID variable length is too long for actual data	Error	1
	SD0021	Missing End Time-Point value	Warning	375
	SD1031	Value for CMENRF is populated, when RFENDTC is NULL	Warning	203
	SD1077	FDA Expected variable not found	Warning	1
	SD1081	CMENDTC variable length is too long for actual data	Warning	1
	SD1081	CMENRF variable length is too long for actual data	Warning	1
	SD1081	CMSPID variable length is too long for actual data	Warning	1
	SD1081	CMSTDTC variable length is too long for actual data	Warning	1
	SD1081	DOMAIN variable length is too long for actual data	Warning	1
	SD1081	STUDYID variable length is too long for actual data	Warning	1
	CT0031	Value for CMROUTE not found in (ROUTE) CT codelist	Notice	16
	CT0049	Value for CMDOSU not found in (UNIT) CT codelist	Notice	19

Figure 21. The Issues Summary - interpreting OpenCDISC

Looking at the first error message (Rule ID: SD0035), we see that the validator found 589 occurrences for a missing value for CMDOSU. Now we are starting to zoom in on the problem. The Details tab provides a description of the SDTM compliance check. Searching for Rule ID SD0035 we quickly determine that the value for our CMDOSU variable is missing and must be populated when –DOSE, --DOSTXT or –DOSTOT is provided.

	A	B	C	D	E
1	Rule ID	Message	Description	Category	Severity
288	SD0033	Missing value for --TPTNUM, when --TPT is provided	Planned Time Point Number (--TPTNUM) should not be NULL, when Planned Time Point Name (--TPT) is populated	Consistency	Warning
289	SD0034	Missing value for --TPTREF, when --ELTM is provided	Time Point Reference (--TPTREF) should not be NULL, when Planned Elapsed Time from Time Point Ref (--ELTM) is populated	Consistency	Warning
290	SD0035	Missing value for --DOSU, when --DOSE, --DOSTXT or --DOSTOT is provided	Dose Units (--DOSU) must be populated, when Dose per Administration (--DOSE), Dose Description (--DOSTXT) or Total Daily Dose (--DOSTOT) is provided	Consistency	Error
291	SD0036	Missing value for --STRESC, when --ORRES is provided	Character Result/Finding in Std Units (--STRESC) must be populated, when Result or Finding in Original Units (--ORRES) is provided	Consistency	Error
292	SD0037	Value for -- not found in (--) user-defined codelist	Variable values should be populated with terms found in the user-defined codelist associated with the variable in define.xml	Terminology	Error

Figure 22. The Rules tab - interpreting OpenCDISC

Now knowing what the problem is, let's look at our SDTM CM file:

[cm]

Window Help

MAIN	USUBJID	CMSEQ	CMSPID	CMTRT	CMDECOD	CMCAT	CMINDC	CMDOSE	CMDOSTXT	CMDOSU	CMROUTE
	PSUG2014-40010007	1	1	WATCHFUL		Cancer Tr		.			
	PSUG2014-40010006	3	1	external		Non-medic	prophylac	.			
	PSUG2014-40010007	2	1	marevan	WARFARIN	Concomita	ablation	.	7.5 mg		ORAL
	PSUG2014-40010007	4	3	zitromax	AZITHROMY	Concomita	infection	.	500 mg		ORAL
	PSUG2014-40010007	3	2	PARACETAM	PARACETAM	Concomita	MUSCLEPAI	.	1000 MG 3		ORAL
	PSUG2014-40010006	1	1	PROSTATEC		Cancer Tr		.			

Figure 23. Dataset screen shot - interpreting OpenCDISC

Here we can see that subject 40010007 is missing CMDOSU when CMDOSTXT is actually populated. And the rule states that CMDOSU should not be blank if CMDOSTXT is populated. So now we can go back to the production program and make the correction or contact the data management team with a potential data issue so that it can be corrected in the SDTM data.

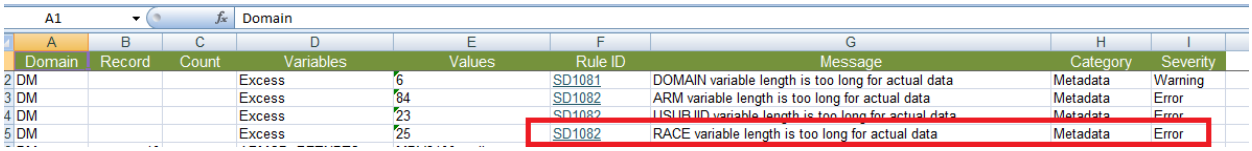
The Rules tab is a for-your-information tab. It reports all of the validator rules by ID and provides the message, description, category and type for each one.

Let's look at another error message. This one is located in the DM file:

DM				
	SD1082	ARM variable length is too long for actual data	Error	1
	SD1082	RACE variable length is too long for actual data	Error	1
	SD1082	USUBJID variable length is too long for actual data	Error	1
	SD0088	RFENDTC is not provided for a randomized subject	Warning	45
	SD1081	ARMCD variable length is too long for actual data	Warning	1
	SD1081	RPTHDTC variable length is too long for actual data	Warning	1

Figure 24. Investigating an error in the DM file - interpreting OpenCDISC

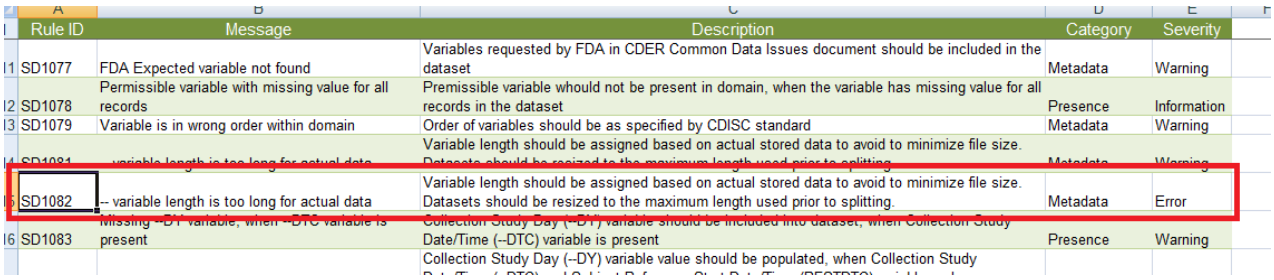
We can see that the DM variable RACE is too long for the actual data and by reviewing the Details tab, we find that the variable length can be shortened by 25 characters (Variables: Excess and Values: 25).



Domain	Record	Count	Variables	Values	Rule ID	Message	Category	Severity
DM			Excess	6	SD1081	DOMAIN variable length is too long for actual data	Metadata	Warning
DM			Excess	84	SD1082	ARM variable length is too long for actual data	Metadata	Error
DM			Excess	23	SD1082	USUBJID variable length is too long for actual data	Metadata	Error
DM			Excess	25	SD1082	RACE variable length is too long for actual data	Metadata	Error

Figure 25. The Details Summary - interpreting OpenCDISC

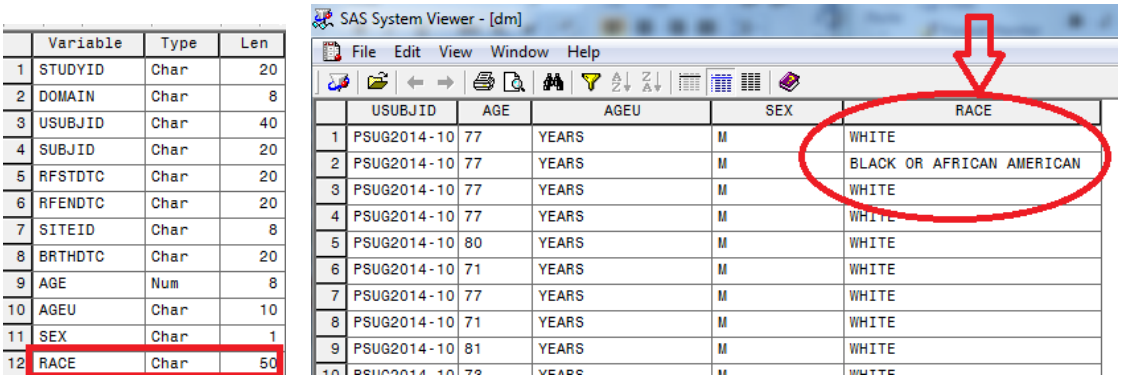
To keep file size to a minimum, CDISC standards suggest that all variables should be assigned a length based on the actual data. This CDISC standard for this rule is outlined on the Rules tab (Figure 26).



Rule ID	Message	Description	Category	Severity
SD1077	FDA Expected variable not found	Variables requested by FDA in CDER Common Data Issues document should be included in the dataset	Metadata	Warning
SD1078	Permissible variable with missing value for all records	Permissible variable whould not be present in domain, when the variable has missing value for all records in the dataset	Presence	Information
SD1079	Variable is in wrong order within domain	Order of variables should be as specified by CDISC standard	Metadata	Warning
SD1081	-- variable length is too long for actual data	Variable length should be assigned based on actual stored data to avoid to minimize file size. Datasets should be resized to the maximum length used prior to splitting.	Metadata	Warning
SD1082	-- variable length is too long for actual data	Variable length should be assigned based on actual stored data to avoid to minimize file size. Datasets should be resized to the maximum length used prior to splitting.	Metadata	Error
SD1083	Missing --DY variable, when --DTC variable is present	Collection Study Day (--DY) variable should be included into dataset, when Collection Study Date/Time (--DTC) variable is present	Presence	Warning
		Collection Study Day (--DY) variable value should be populated, when Collection Study		

Figure 26. The Rules tab - interpreting OpenCDISC

Checking the DM dataset, we find that the variable RACE is in fact longer than it needs to be. The maximum length for RACE is about 30 characters and it has been defined as 50. Adjusting the length of the variable RACE will get rid of the error message.



	Variable	Type	Len
1	STUDYID	Char	20
2	DOMAIN	Char	8
3	USUBJID	Char	40
4	SUBJID	Char	20
5	RFSTDTC	Char	20
6	RFENDTC	Char	20
7	SITEID	Char	8
8	BRTHDTC	Char	20
9	AGE	Num	8
10	AGEU	Char	10
11	SEX	Char	1
12	RACE	Char	50

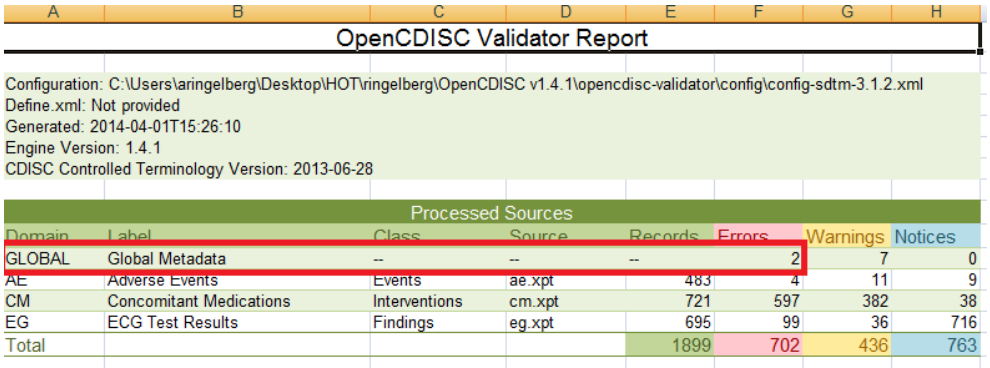
SAS System Viewer - [dm]

File Edit View Window Help

	USUBJID	AGE	AGEU	SEX	RACE
1	PSUG2014-10	77	YEARS	M	WHITE
2	PSUG2014-10	77	YEARS	M	BLACK OR AFRICAN AMERICAN
3	PSUG2014-10	77	YEARS	M	WHITE
4	PSUG2014-10	77	YEARS	M	WHITE
5	PSUG2014-10	80	YEARS	M	WHITE
6	PSUG2014-10	71	YEARS	M	WHITE
7	PSUG2014-10	77	YEARS	M	WHITE
8	PSUG2014-10	71	YEARS	M	WHITE
9	PSUG2014-10	81	YEARS	M	WHITE

Figure 27. Screen shot of the DM file (variable view and formatted view) - interpreting OpenCDISC

Running the validator to check the AE, CM and EG SDTM files, we get the following report:



OpenCDISC Validator Report

Configuration: C:\Users\aringelberg\Desktop\HOT\ringelberg\OpenCDISC v1.4.1\opencdisc-validator\config\config-sdtm-3.1.2.xml
Define.xml: Not provided
Generated: 2014-04-01T15:26:10
Engine Version: 1.4.1
CDISC Controlled Terminology Version: 2013-06-28

Processed Sources

Domain	Label	Class	Source	Records	Errors	Warnings	Notices
GLOBAL	Global Metadata	--	--	--	2	7	0
AE	Adverse Events	Events	ae.xpt	483	4	11	9
CM	Concomitant Medications	Interventions	cm.xpt	721	597	382	38
EG	ECG Test Results	Findings	eg.xpt	695	99	36	716
Total				1899	702	436	763

Figure 28. Global errors - interpreting OpenCDISC

This time, let's focus on the GLOBAL errors, of which there are two. The ISSUES summary tab tells us where the errors are located.

Issue Summary				
Source	Rule ID	Message	Severity	Found
GLOBAL				
	SD1020	Missing DM dataset	Error	1
	SD1115	Missing TS dataset	Error	1
	SD1107	Missing LB dataset	Warning	1
	SD1108	Missing VS dataset	Warning	1
	SD1109	Missing EX dataset	Warning	1
	SD1110	Missing DS dataset	Warning	1
	SD1111	Missing SE dataset	Warning	1
	SD1112	Missing TA dataset	Warning	1
	SD1113	Missing TE dataset	Warning	1

Figure 29. The Issues Summary/Global errors - interpreting OpenCDISC

Investigating the first error that is reflected in the DM file, Rule ID SD1020 tells us that the DM file must be included in every submission. So to get rid of this global error message, include the DM file as one of the files to be checked by the validator. Also, if we investigate the global error on the TS file, you will find that it too must be included in every submission. To get rid of these two errors, make sure that these files are included.

	A	B	C	D	E
1	Rule ID	Message	Description	Category	Severity
362	SD1018	VISITNUM/VISIT/VISITDY values do not match TV domain data	The combination of Visit Number (VISITNUM), Visit Name (VISIT), and Planned Study Day of Visit (VISITDY) values should match entries in the Trial Visits (TV) dataset, when they are planned visits (SVUPDES = NULL)	Cross-reference	Warning
363	SD1019	VISITDY is populated for unplanned visit	Planned Study Day of Visit (VISITDY) should equal NULL for unplanned visits, where Description of Unplanned Visit (SVUPDES) is populated	Consistency	Warning
364	SD1020	Missing DM dataset	Demographics (DM) dataset must be included in every submission	Presence	Error
365	SD1021	Unexpected character value in -- variable	Character values should not have leading space ' ' characters, or ' ' as an entire value. The only exceptions are COVALn and TSVALn variables.	Format	Warning
366	SD1022	Invalid value for QNAM variable	The value of Qualifier Variable Name (QNAM) should be limited to 8 characters, cannot start with a number, and cannot contain characters other than letters in upper case, numbers, or underscores	Format	Warning

Figure 30. The Rules tab/Global errors - interpreting OpenCDISC

Let's look at one more error. This one is found in the EG file:

EG				
	CT0085R	EGTEST and EGTESTCD values do not have the same Code in CDISC CT	Error	92
	SD1082	EGCAT variable length is too long for actual data	Error	1
	SD1082	EGEVAL variable length is too long for actual data	Error	1
	SD1082	EGMETHOD variable length is too long for actual data	Error	1
	SD1082	EGORRES variable length is too long for actual data	Error	1

Figure 31. The EG file/Global errors - interpreting OpenCDISC

Rule ID CT085R refers to an issue with the code list. The details tab explains what the problem is:

	A	B	C	D	E
1	Rule ID	Message	Description	Category	Severity
89	CT0085R	EGTEST and EGTESTCD values do not have the same Code in CDISC CT	In ECG Test Results (EG) domain values for ECG Test or Examination Short Name (EGTESTCD) and ECG Test or Examination Name (EGTEST) variables must be populated using terms with the same Codelist Code value in CDISC control terminology. There is one-to-one relationship between EGTESTCD and EGTEST values defined in CDISC control terminology by Codelist Code value.	Terminology	Error

Figure 32. The Rule ID for the CT0085R/Global errors - interpreting OpenCDISC

Our EG file contains the following values for EGTEST and EGTESTCD:

SAS System Viewer - [eg]

File Edit View Window Help

	EGTESTCD	EGTEST
1	ABNORM	ECG Abnormality
2	ABNORM	ECG Abnormality
3	ABNORM	ECG Abnormality
4	ABNORM	ECG Abnormality
5	ABNORM	ECG Abnormality
6	ABNORM	ECG Abnormality
7	ABNORM	ECG Abnormality
8	ABNORM	ECG Abnormality
9	ABNORM	ECG Abnormality
10	ABNORM	ECG Abnormality
11	ABNORM	ECG Abnormality
12	ABNORM	ECG Abnormality
13	ABNORM	ECG Abnormality
14	ABNORM	ECG Abnormality
15	HRMEAN	Summary (Mean) Heart Rate
16	HRMEAN	Summary (Mean) Heart Rate
17	HRMEAN	Summary (Mean) Heart Rate
18	HRMEAN	Summary (Mean) Heart Rate
19	HRMEAN	Summary (Mean) Heart Rate
20	HRMEAN	Summary (Mean) Heart Rate
21	HRMEAN	Summary (Mean) Heart Rate
22	HRMEAN	Summary (Mean) Heart Rate
23	HRMEAN	Summary (Mean) Heart Rate
24	INTP	ECG Interpretation
25	INTP	ECG Interpretation
26	INTP	ECG Interpretation

Figure 33. EG data screen shot/Global errors - interpreting OpenCDISC

Searching through the code list, we see that EGTESTCD=ABNORM is not in the list. Now the decision to keep the variable as is, or make a correction needs to be made. In this example, we ended up keeping EGTESTCD=ABNORM as part of the EG domain and ignored the error message.

CDISC SDTM Controlled Terminology

	U WAVE ABNORMALITY		C106579	Abnormal U Wave
Back to top				
CL.C71153.EGTESTCD	ECG Test Code (EGTESTCD)	text Extensible: Yes	C71153	ECG Test Code
	HRMAX		C39779	Summary (Max) Heart Rate
	PRMEAN		C62086	Summary (Mean) PR Duration
	QRSDUR		C62087	Summary (Mean) QRS Duration

This is just a snap shot of part of the EGTESTCD SDTM codelist.

Figure 34. CDISC SDTM Controlled Terminology - interpreting OpenCDISC

CUSTOMIZING THE OPENCDISC VALIDATOR

There may be occasions when the default OpenCDISC validator options are not enough or appropriate. Our team recently ran across a situation where they were using an amended version of SDTM. In order to properly validate the SDTM files, the team had to track down and load files from a prior release and copy them into the OpenCDISC install directory before they could properly validate their SDTM files.

Rules may be added, modified and/or deleted, MedDRA versions may be changed, and CDISC CT selected to whatever your specifications are. The details for customizing OpenCDISC are beyond the scope of this paper, but keep in mind that it is customizable and you can create a validator that suits your needs. For more information on customizing the validation framework, check out the OpenCDISC site: http://www.opencdisc.org/projects/validator/opencdisc-validation-framework.

CREATING DEFINE.XML

The OpenCDISC Validator can also generate DEFINE.XML. Don't get too excited, the validator is only able to generate a DEFINE.XML shell. This is because the DEFINE.XML is created using the SDTM metadata and not all of the information needed for the document is found in the metadata. However, it is worth seeing what the validator can do for you.

From the OpenCDISC Validator window perform the following steps.

Step 1: From the 'What would you like to do?' prompt, choose 'Generate Define.xml'.

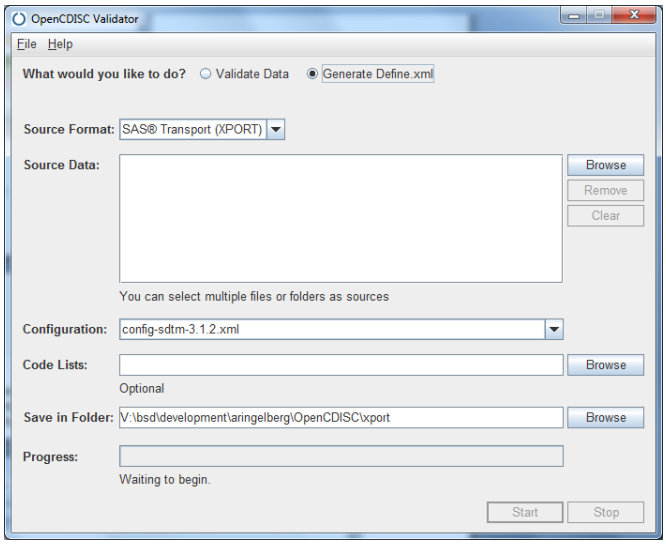


Figure 35. The OpenCDISC interface - Generate Define.xml

Step 2: Choose the Source Format. For this example we are using the default, XPORT.

Step 3: Choose the file(s) you want to create define.xml for using the Browse button.

Step 4: Choose the Configuration. We are using 'config-sdtm-3.1.2.xml' for this example.

Step 5: Select the folder where you want to save the define.xml file.

Step 6: Click on Start.

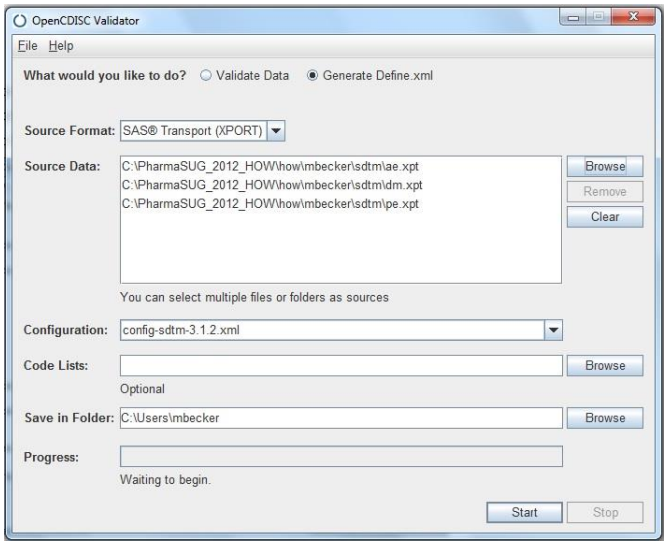


Figure 36. The OpenCDISC interface file selection - Generate Define.xml

When the define.xml file has been generated, the following information screen will appear:

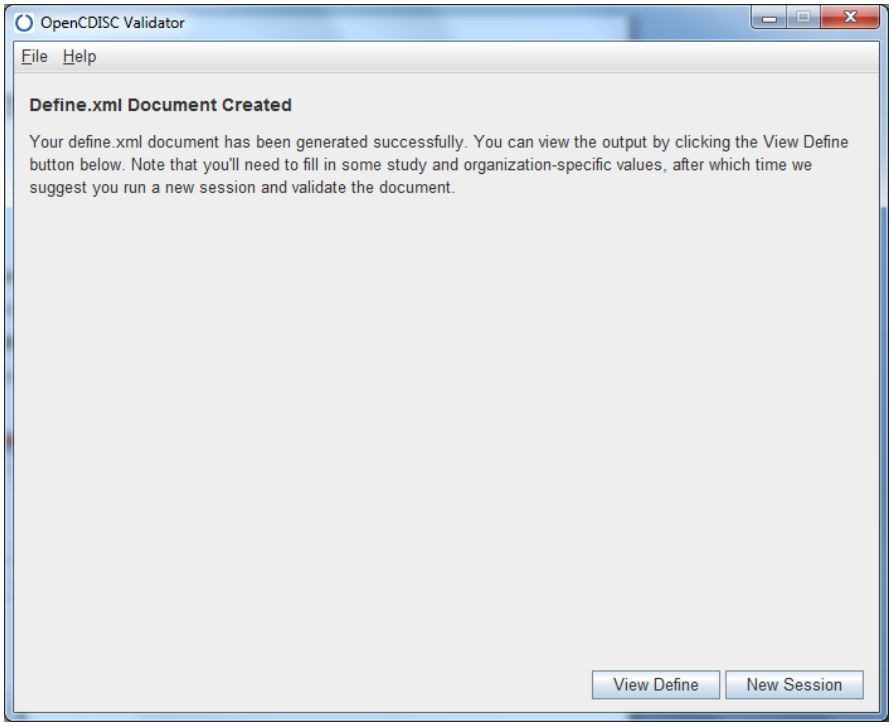


Figure 37. The OpenCDISC summary report - Generate Define.xml

Step 7: Choose 'View Define' to see the newly created define.xml file.

In this example, 3 SDTM files were selected: DM, AE and PE. Below is some of the XML output. Note that some study and organization-specific values still need to be filled in.

Datasets for Study

Dataset	Description	Structure	Purpose	Keys	Location
DM	Demographics	Special Purpose - One record per subject	Tabulation		dm.xpt
AE	Adverse Events	Events - One record per event per subject	Tabulation		ae.xpt
PE	Physical Examination	Findings - One record per event per subject	Tabulation		pe.xpt

Go to the top of the define.xml

Date of document generation (2012-04-24T12:25:05)

Demographics Dataset (DM)

Variable	Label	Type	Controlled Terms or Format	Origin	Role	Comment
USUBJID	Unique Subject Identifier	text			Identifier	
STUDYID	Study Identifier	text			Identifier	
DOMAIN	Domain Abbreviation	text			Identifier	
SUBJID	Subject Identifier for the Study	text			Topic	
RFSTDTC	Subject Reference Start Date/Time	datetime			Record Qualifier	
RFENDTC	Subject Reference End Date/Time	datetime			Record Qualifier	
SITEID	Study Site Identifier	text			Record Qualifier	
BRTHDTC	Date/Time of Birth	datetime			Record Qualifier	
AGE	Age	integer			Record Qualifier	
AGEU	Age Units	text			Variable Qualifier	
SEX	Sex	text			Record Qualifier	
RACE	Race	text			Record Qualifier	
ETHNIC	Ethnicity	text			Record Qualifier	
ARMCD	Planned Arm Code	text			Record Qualifier	
ARM	Description of Planned Arm	text			Synonym Qualifier	
COUNTRY	Country	text			Record Qualifier	

Go to the top of the define.xml

Date of document generation (2012-04-24T12:25:05)

Figure 38. The Define.xml - Generate Define.xml

Adverse Events Dataset (AE)

Variable	Label	Type	Controlled Terms or Format	Origin	Role	Comment
USUBJID	Unique Subject Identifier	text			Identifier	
STUDYID	Study Identifier	text			Identifier	
DOMAIN	Domain Abbreviation	text			Identifier	
AESEQ	Sequence Number	float			Identifier	
AEREFID	Reference ID	text			Identifier	
AETERM	Reported Term for the Adverse Event	text			Topic	
AEMODIFY	Modified Reported Term	text			Synonym Qualifier	
AEDECOD	Dictionary-Derived Term	text			Synonym Qualifier	
AEBODSYS	Body System or Organ Class	text			Record Qualifier	
AESER	Serious Event	text			Record Qualifier	
AEACN	Action Taken with Study Treatment	text			Record Qualifier	
AEREL	Causality	text			Record Qualifier	
AEOUT	Outcome of Adverse Event	text			Record Qualifier	
AESHOSP	Requires or Prolongs Hospitalization	text			Record Qualifier	
AECONTRT	Concomitant or Additional Trtmnt Given	text			Record Qualifier	
AETOXGR	Standard Toxicity Grade	text			Record Qualifier	
AESTDTC	Start Date/Time of Adverse Event	datetime			Timing	
AEENDTC	End Date/Time of Adverse Event	datetime			Timing	
AEENRF	End Relative to Reference Period	text			Timing	

Go to the top of the define.xml

Date of document generation (2012-04-24T12:25:05)

Physical Examination Dataset (PE)

Variable	Label	Type	Controlled Terms or Format	Origin	Role	Comment
USUBJID	Unique Subject Identifier	text			Identifier	
STUDYID	Study Identifier	text			Identifier	
DOMAIN	Domain Abbreviation	text			Identifier	
PESEQ	Sequence Number	float			Identifier	
PETESTCD	Body System Examined Short Name	text			Topic	
PETEST	Body System Examined	text			Synonym Qualifier	

Figure 39. The Define.xml (con't) - Generate Define.xml

VALIDATING DEFINE.XML

Once Define.xml has been created, either by using the OpenCDISC validator or by another means, the OpenCDISC validator can actually be used in validating the define.xml. The same basic validation process that is used with SDTM and ADaM is applied to the Define.xml as well.

Step 1: From the 'What would you like to do?' prompt, choose 'Validate Data'

Step 2 : Select Define.xml as the Standard

Step 3: Use the Browse button to point to the location of the define.xml file

Step 4: Your configuration will automatically come up as config-define-1.0.xml

Step 5: Chose your report format

Step 6: Click on the Start button to run the report

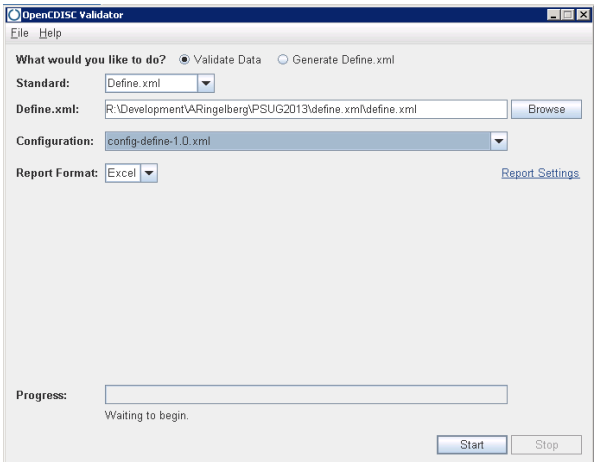
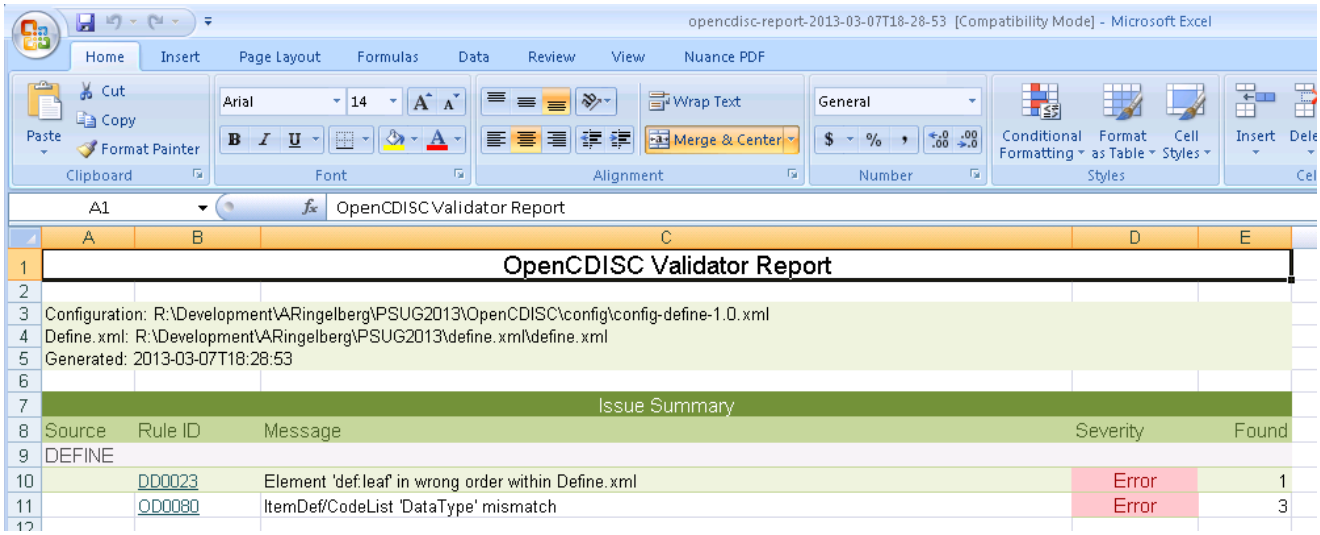


Figure 40. The OpenCDISC interface - Validate Define.xml

Step 7: As in the validation examples for SDTM and ADaM, select View Report

Figure 41. The validation summary report - Validate Define.xml

Step 8: Look over the Open CDISC Validator Report reviewing the Issue Summary tab, the Details tab and Rules tab as needed to determine what problems exist and where to make the correction in your Define.xml.



OpenCDISC Validator Report

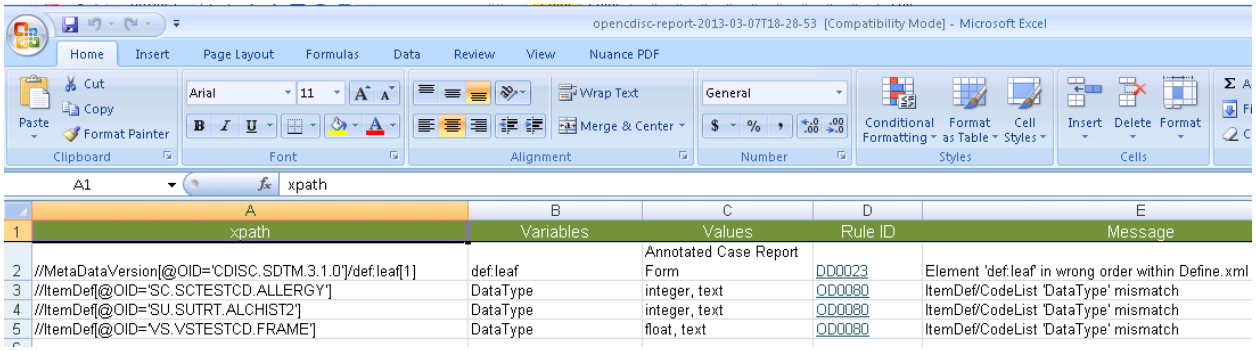
Configuration: R:\Development\ARingelberg\PSUG2013\OpenCDISC\config\config-define-1.0.xml
Define.xml: R:\Development\ARingelberg\PSUG2013\define.xml\define.xml
Generated: 2013-03-07T18:28:53

Issue Summary

Source	Rule ID	Message	Severity	Found
DEFINE				
	DD0023	Element 'def:leaf' in wrong order within Define.xml	Error	1
	OD0080	ItemDef/CodeList 'DataType' mismatch	Error	3

Figure 42. The Issues Summary - Validate Define.xml

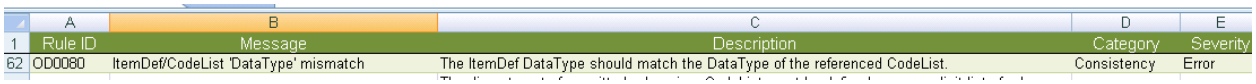
For example, in the details tab there is an error on line 3 for Rule OD0080. XPATH shows where the issue is located: 'SC. SCTESTC.ALLERGY'.



xpath	Variables	Values	Rule ID	Message
		Annotated Case Report		
//MetaDataVersion[@OID='CDISC.SDTM.3.1.0']/def:leaf[1]	def:leaf	Form	DD0023	Element 'def:leaf' in wrong order within Define.xml
//ItemDef[@OID='SC.SCTESTCD.ALLERGY']	DataType	integer, text	OD0080	ItemDef/CodeList 'DataType' mismatch
//ItemDef[@OID='SU.SUTRT.ALCHIST2']	DataType	integer, text	OD0080	ItemDef/CodeList 'DataType' mismatch
//ItemDef[@OID='VS.VSTESTCD.FRAME']	DataType	float, text	OD0080	ItemDef/CodeList 'DataType' mismatch

Figure 43. The Details tab - Validate Define.xml

Reviewing the rule, it is determined that the data type should match the code list.



Rule ID	Message	Description	Category	Severity
OD0080	ItemDef/CodeList 'DataType' mismatch	The ItemDef DataType should match the DataType of the referenced CodeList.	Consistency	Error

Figure 44. The Rules tab - Validate Define.xml

Looking at the code list in the Define.xml file, ALLERGY is defined as type 'integer' with a format defined by YESNOUNK

Value Level Metadata

Source Variable	Value	Label	Type	Controlled Terms or Format	Origin	Role	Comment
SCTESTCD	ALLERGY	Allergy Status	integer	YESNOUNK	CRF Page		Subject Characteristics CRF Page 4
SCTESTCD	EDLEVLN	EDUCATIONAL LEVEL-DVN	float		CRF Page		Subject Characteristics CRF Page 4
SCTESTCD	EXCLSN	EXERCISE CLASSIFICATION-DVN	float		CRF Page		Subject Characteristics CRF Page 4
SCTESTCD	RACEOTH	Text for other race	text		CRF Page		Subject Characteristics CRF Page 4

Figure 45. SCTESTCD in Define.xml - Validate Define.xml

The validator is expecting to find an integer value. First check the value of ALLERGY. If it is text, then the Type should be changed to 'text' in the Define.xml. Also look at the format that is associated with the variable.

YESNOUNK is defined as text ('Y' 'N' 'U' 'NA'). This is another clue that Type is not correctly defined for the variable ALLERGY.

YESNOUNK, Reference Name (YESNOUNK)	
Y	YES
N	NO
U	UNKNOWN
NA	NOT APPLICABLE

Figure 46. YESNOUNK controlled terminology - Validate Define.xml

Once the Define.xml has been updated, you can run the new version through the validator again. When the report no longer generates error and or warning messages, the define.xml is CDISC compliant.

CONCLUSION

The development of the open-source tool, the OpenCDISC Validator, is helping to ensure data compliance with CDISC models such as SDTM, ADaM and Define.xml. The tool is of commercial-quality, freely available, up to date with most recent CDISC standards and is EASY to use. This tool reduces the effort for individualized quality control in order to make sure SDTM and ADaM files are CDISC compliant. The information generated by the validator is easy to interpret once you are familiar with CDISC conventions andallows the user to make corrections in the affected file - bringing that file up to CDISC standards. The validator also has the ability to create a define.xml shell as a starting point and also validates a fully populated define.xml. It is as simple as bringing up the OpenCDISC Validator, choosing the option of 'Create define.xml' and instructing the validator which SDTM files need to be included. It is hoped that this paper will demonostrate how easy the OpenCDISC Validator is to use while at the same time saving valuable time and effort when it comes to the validation of SDTM, ADaM and define.xml.

ACKNOWLEDGMENTS

The authors would like to thank our employer inVentiv Health Clinical for supporting our participation in PharmaSUG, along with all our friends at inVentiv, especially Brian Fairfield-Carter for being our editor in chief.

REFERENCES

HTTP://WWW.CDISC.ORG

HTTP://WWW.OPENCDISC.ORG

CONTACT INFORMATION

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

Name: Angela Ringelberg
E-mail: aringelberg@ockham.com

Name: Tracy Sherman
E-mail: tracy.sherman@inventivhealth.com