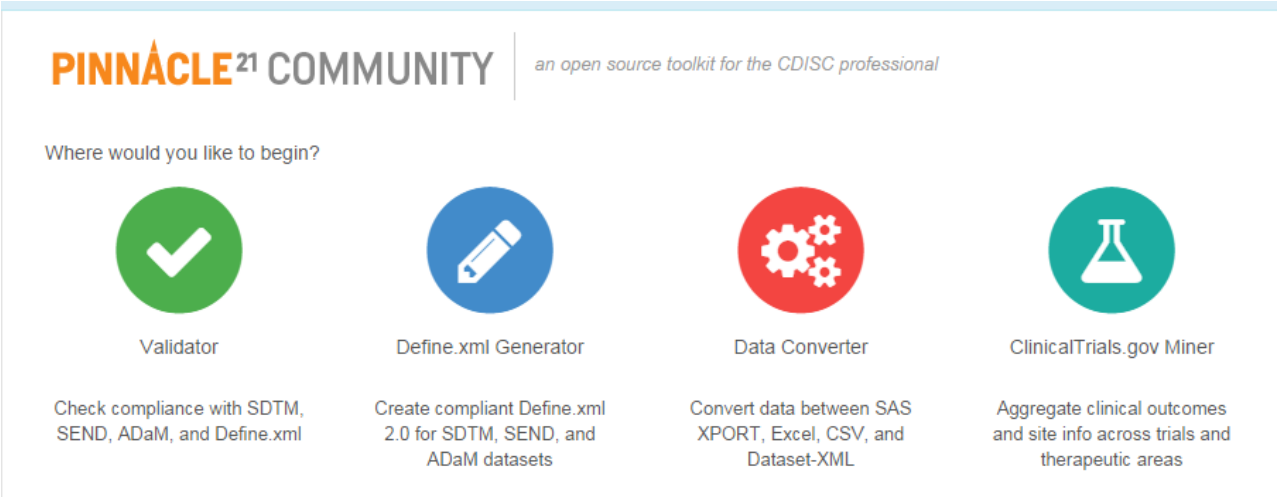
Running CDISC Conformance Checking in SAS

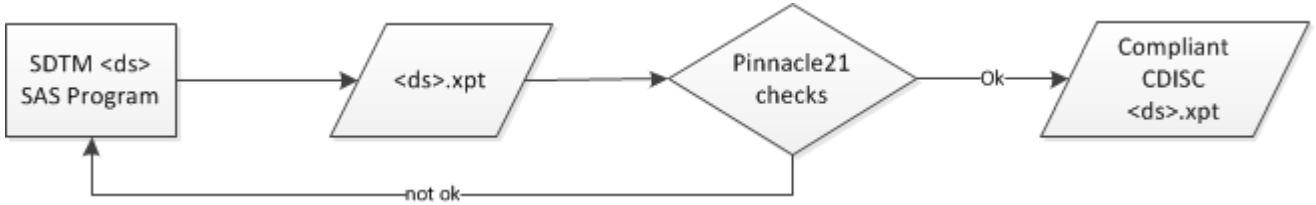
or how to run Pinnacle 21 in SAS

How to check against a global SDTM standard and their rules?

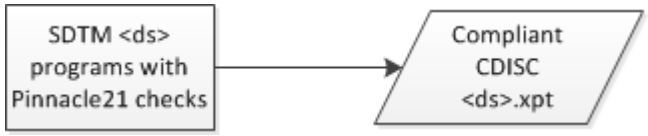
Use Pinnacle 21 but not in the interactive mode.



Dataset creation and CDISC compliance check – two step approach



Dataset creation and CDISC compliance check – one step approach



1. How to run Pinnacle21 in SAS

First Option: Call validator-cli-<ver.>.jar in command line

- Path: <pinnacle21 install dir>\components\lib
- Parameter Reference:

Parameter Type	Parameter	Valid Values	Default	Applicable Task Type	Description
	-type	SDTM,ADaM, SEND,Define,Custom	SDTM	Validate	Data Standard/Structure to validate
Source Data	-source	<path>		both	Path to the source data files
	-source:type	SAS|Delimited	SAS	both	file format
	-source:delimiter	<delimiter>	,	both	field Delimeter if source:type = Delimited
	-source:qualifier	<qualifier>	"	both	field qualifier if source:type = Delimited
Configuration	-config	<path>		both	path the xml configuration document which lists rules to validate
	-config:define	<path>		Validate	path and name of define.xml for the study
	-config:cdisc	<version>		Validate	CDISC Controlled Terminology Version
	-config:meddra	<version>		Validate	MedDRA dictionary version
Report	-report	<path>		Validate	path and file name to where the report will be saved
	-report:type	Excel,CSV,XML	Excel	Validate	report format
	-report:cutoff	<#>	1000	Validate	Number of times a record detail is printed per distinct domain issue
	-report:overwrite	yes,no		Validate	indicates if report should be overwritten if same name provided

Link: https://www.pinnacle21.com/using-opencdisc-validator-cli

- Command Line Example:

java -Xms256m -Xmx1024m -jar validator-cli-2.1.2.jar -type=SDTM -source=/home/user/opencdisc-validator/data/6073K1-006/*.XPT -source:type=SAS -config=/home/user/opencdisc-validator/config/config-sdtm-3.1.1.xml -config:define=/home/user/opencdisc-validator/data/6073K1-006/define.xml -config:cdisc=2012-12-21 -config:meddra=14.0 -report=/home/user/opencdisc-validator/reports/Drug-study-report1000.xls -report:type=Excel -report:cutoff=1000 report:overwrite=Yes

First Option: Call validator-cli-<ver.>.jar in command line

- Call the validator-cli jar file in SAS via command line parameters

```
15
16 DATA _null_;
17      CALL SYSTEM("java -Xms256m -Xmx1024m -jar components/lib/validator-cli-2.1.2.jar -type=SDTM -source=C:/ .....";
18 RUN;
19
```

- Full example

```
21
22 DATA _null_;
23      /* call the validator-cli jar file */
24      /* java = call Java, -Xms = start amount of memory, -Xmx = maximum amount of memory -jar = path+jar file */
25      CALL SYSTEM(java -Xms256m -Xmx1024m -jar <path>/components/lib/validator-cli-2.1.2.jar
26          /* check SDTM */
27          -type=SDTM
28          /* source data path single xpt/csv or all (*) datasets */
29          -source=<path>/data/nicardipine/nicsah1/sdtm/xpt/*.xpt
30          /* source type SAS/Delimited (default SAS) */
31          -source:type=SAS
32          /* more available source paramters: delimiter, qualifier */
33          /* XML list of rules to validate */
34          -config=<path>/components/config/SDTM_3.1.3_(FDA).xml
35          /* CDISC Controlled Terminology Version <SDTM path> = components\config\ds
36          -config:cdisc=2014-09-26
37          /* more available config parameters: define = path+file, meddra = version
38          or extracted MedDRA dict (Preferred Term) is stored under components/confi
39          /* report path+file */
40          -report=<path>/output/sdtm_check_report.xlsx
41          /* report type = Excel (default), CSV, XML */
42          -report:type=Excel
43          /* report cutoff value */
44          -report:cutoff=1000
45          /* report overwrite = yes/no */
46          -report:overwrite=Yes
47      );
48 RUN;
49
```

Disadvantage
- all parameters in one line
- automization is difficult

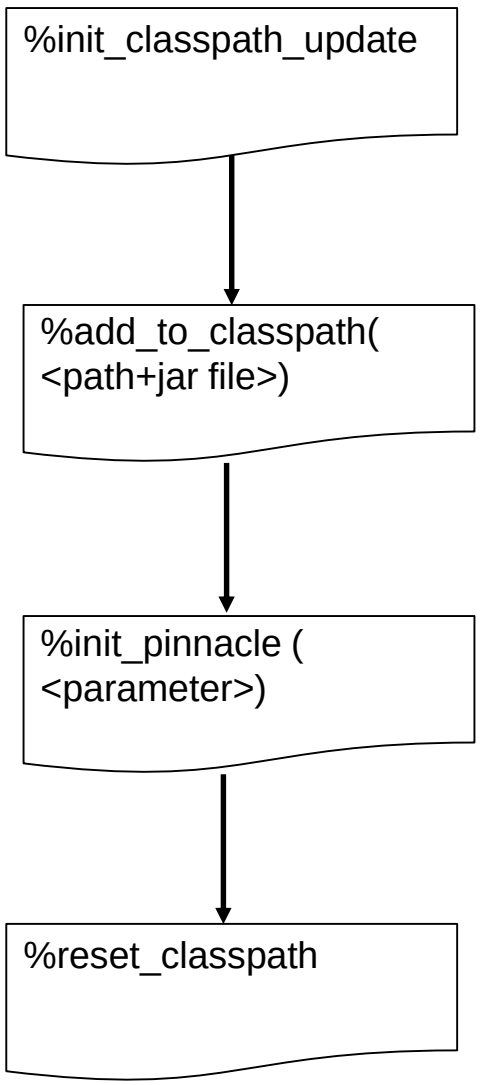
Second Option: Call validator-cli-<version>.jar in SAS

- Check your Java system settings (system variables)
  - JAVA_HOME = C:\Program Files(x86)\Java\jdk1.7.0_79 -> *Location of the Java Development Kit or Java Runtime Environment*
  - Expand the Path = <path>;%JAVA_HOME%\bin -> *Location of the java/javaws.exe*
  - CLASSPATH = . -> *The variable is one way to tell applications, including the JDK tools, where to look for user classes*

- Three SAS macros are required
  - %init_classpath_update = initialize the Java class path
  - %add_to_classpath(<path>/<jar file>) = add jar file to the Java class path
  - %reset_classpath = reset to the original Java class path
- One self created SAS macro is required
  - %init_pinnacle = split parameters and call the jar file

- More information
  - PATH and CLASSPATH: https://docs.oracle.com/javase/tutorial/essential/environment/paths.html
  - SAS Java Macros: *Sample 38518: Dynamically setting the CLASSPATH for use in a DATA step that uses the Java object component http://support.sas.com/kb/38/518.html*

Second Option: Call validator-cli-<version>.jar in SAS



Read-out the CLASSPATH variable and the path separator information

Create global macro variables:
CP_orig_classpath
CP_path_separator

Extend the Java CLASSPATH variable

Required global macro variable:
CP_orig_classpath
CP_path_separator
from macro init_classpath_update

Call the added jar file

Reset CLASSPATH variable

Required global macro variable:
CP_orig_classpath
from macro init_classpath_update

Second Option: Call validator-cli-<version>.jar in SAS

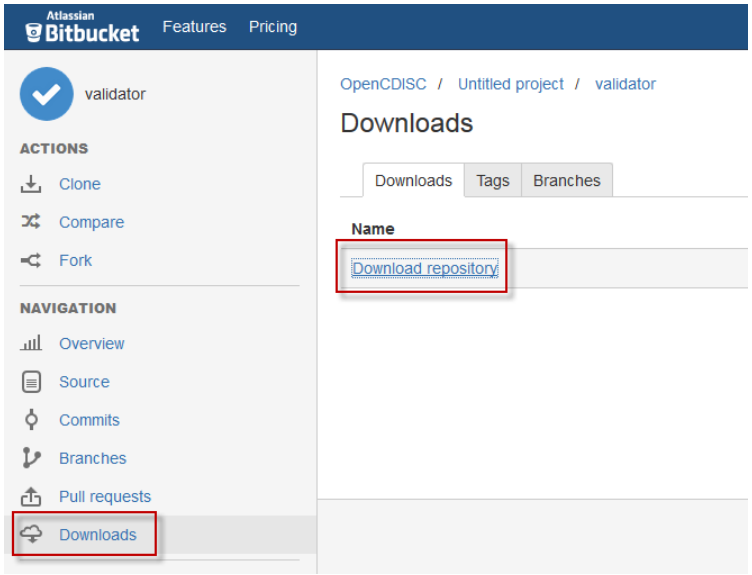
- Example

```
50
51 /* add needed macros */
52 OPTION mautosource sasautos=("macros" sasautos);
53
54
55 /* set macro variables */
56 %LET _root=%STR(C:/grt/cdisc_comp);
57 %LET _study_sdtm=%STR(&_root./data/nicardipine/nicsah1/sdtm/xpt);
58 %LET _output_sdtm=&_root./output/check_results_sdtm_%SYSFUNC(TRANSLATE(%SYSFUNC(DATETIME(), E8601DT.),'-',':')).xlsx;
59
60
61 /* initialize the Java class path */
62 %init_classpath_update;
63
64
65 /* add the pinnacle21 jar file to the Java class path */
66 %add_to_classpath(C:/grt/cdisc_comp/pinnacle21-community/components/lib/validator-cli-2.1.2.jar);
67
68
69 /* validate nicash SDTM datasets */
70 %init_pinnacle(params=%NRBQUOTE(
71                task=Validate,
72                source=&_study_sdtm.,
73                config=&_root./pinnacle/opencdisc-community/components/config/SDTM_3.1.3_(FDA).xml,
74                config:cdisc=2014-09-26,
75                report=&_output_sdtm.,
76                report:type=Excel,
77                report:cutoff=10,
78                report:overwrite=yes),
79                debug=N);
80
81
82 /* reset to the original Java class path */
83 %reset_classpath;
84
```

2. Create pinnacle21 jar file from source

Create pinnacle21 jar file from source

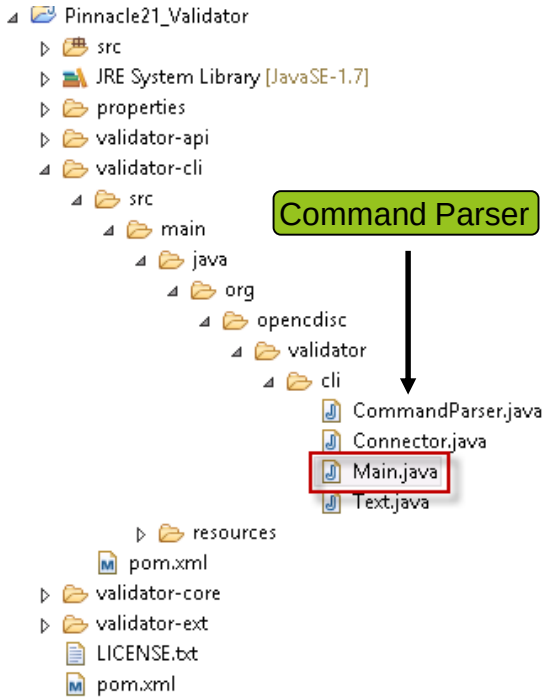
- Execute pinnacle21 in an Linux/Unix SAS or an SAS hosted environment
- Preparation:
  - Downlod the code repository

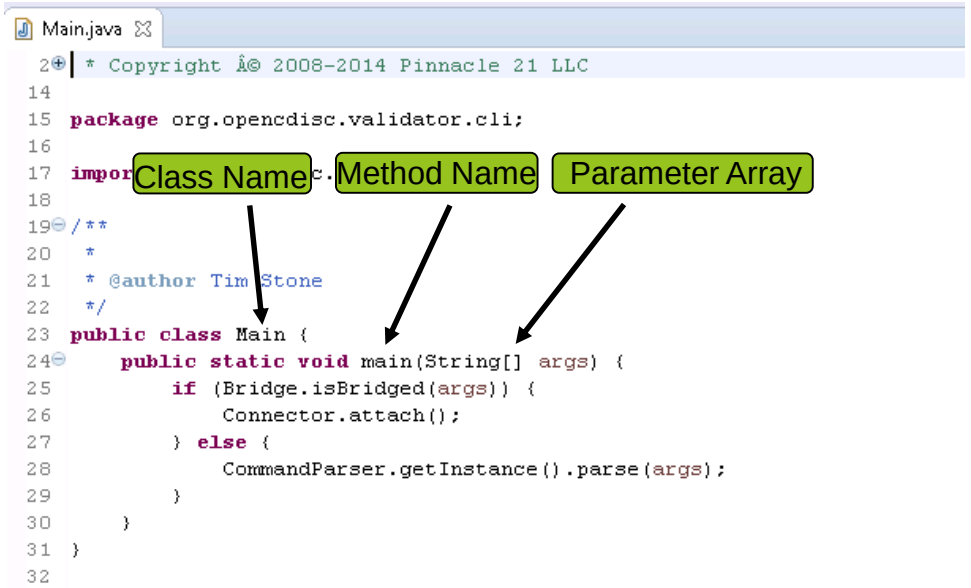


Link: https://bitbucket.org/opencdisc/validator/downloads/

Create pinnacle21 jar file from source

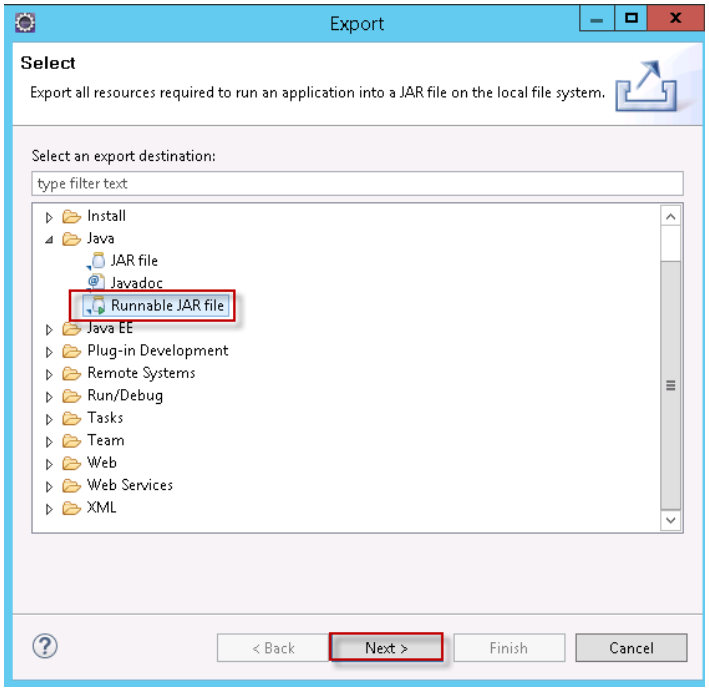
- Import the file into Eclipse

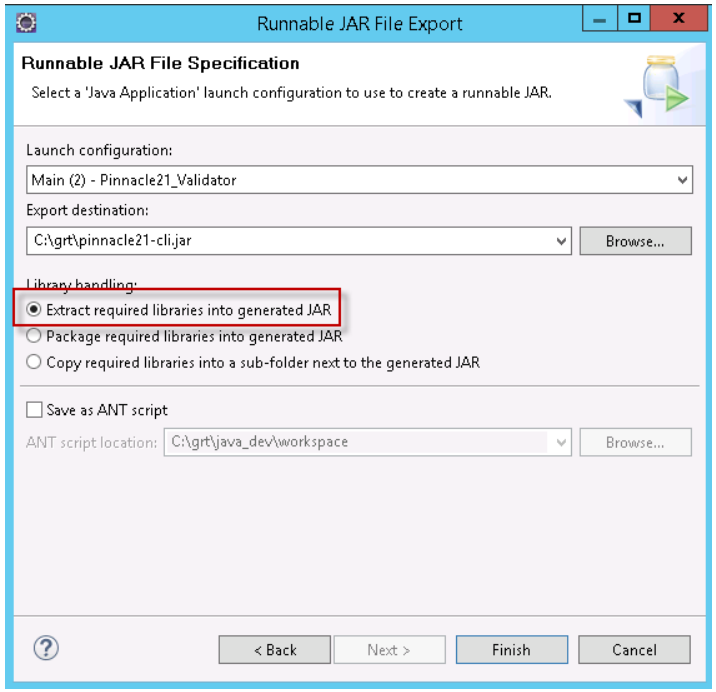




Create pinnacle21 jar file from source

- Create an JAR file





Create pinnacle21 jar file from source

- Call the JAR file

```
1
2  /* add needed macros */
3  OPTION mautosource sasautos=("macros" sasautos);
4
5  %LET _root=%STR(C:/grt/cdisc_comp);
6
7  %LET _study_sdtm=%STR(&_root./data/nicardipine/nicsah1/sdtm/xpt);
8  %LET _output_sdtm=&_root./output/check_results_sdtm_%SYSFUNC(TRANSLATE(%SYSFUNC(DATETIME(), E8601DT.),'-',':')).xlsx;
9
10 /* initialize the Java class path */
11 %init_classpath_update;
12
13 /* add the pinnacle21 jar file to the Java class path */
14 %add_to_classpath(&_root./jar/pinnacle21-cli.jar);
15
16 /* validate nicash SDTM datasets */
17 %init_pinnacle(params=%NRBQUOTE(
39
40     DATA work._opencdisc_cli;
41         dcl javaobj j("org/opencdisc/validator/cli/Main");/* folder structure in the JAR file */
42         array s{&_i.} $200 (&_cmd.);
43         j.callStaticVoidMethod("main",s);/* call main method */
44     RUN;
45
26                 debug=N);
27
28 /* reset to th eoriginal Java class path */
29 %reset_classpath;
30
31 /* end of file */
```

Class Main

String [ ] args

public static void main

References

- SAS Java macros:
  - http://support.sas.com/kb/38/518.html
- Using OpenCDISC Validator CLI:
  - https://www.pinnacle21.com/using-opencdisc-validator-cli
- Pinnacle21 BitBucket repository
  - https://bitbucket.org/opencdisc/validator/overview
- Pinnacle21 SVN repository
  - http://svn.opencdisc.org/validator/ (accessible with SVN)
- SAS DECLARE Statement, Java Object
  - https://support.sas.com/documentation/cdl/en/lecompobjref/69740/HTML/default/viewer.htm#n0cdvh1d0v4g5rn0z9dnihwtatnx.htm

Questions?

Many thanks for your attention!

Frank Biedermann

Analytical Programmer

Contact
phone +49 241 569 2462
email frank.biedermann@grunenthal.com

Grünenthal GmbH
Zieglerstr. 6 • 52078 Aachen

www.grunenthal.com

GRÜNENTHAL