

The Concept of IoT - The Next Generation of Statistical Computing Platform

Zhiping Yan, Huadan Li

Nov 2021



Background and Rationale

Issue

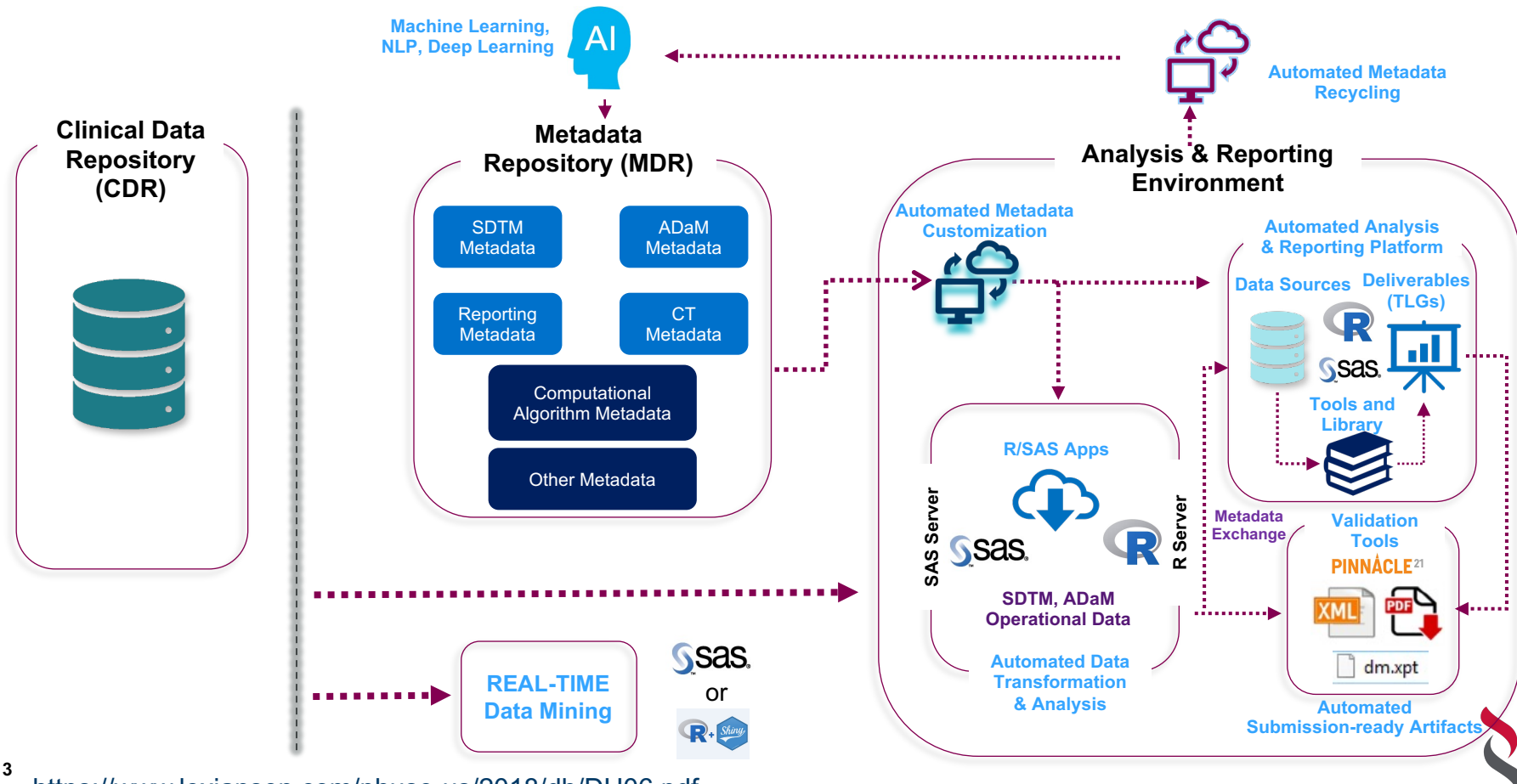
- Toggle between software, folders and files kills productivity
- Connecting all components and bringing them to the front of desktop within short time is urgent

Solution

- Concept of IoT can be borrowed to integrate programming activities into an ecosystem
- Python allows different software programs to be interoperated and integrated into one platform
- Python GUI toolkit intended to develop Graphical User Interface can make operations and interaction more intuitive and easier to use and learn
- At Dizal, an Integrated Statistical Computing Platform (hereinafter referred to as Dzial-iSCP) allowing users to access standalone applications is built to provide a scalable, portable, and extensible operating environment.



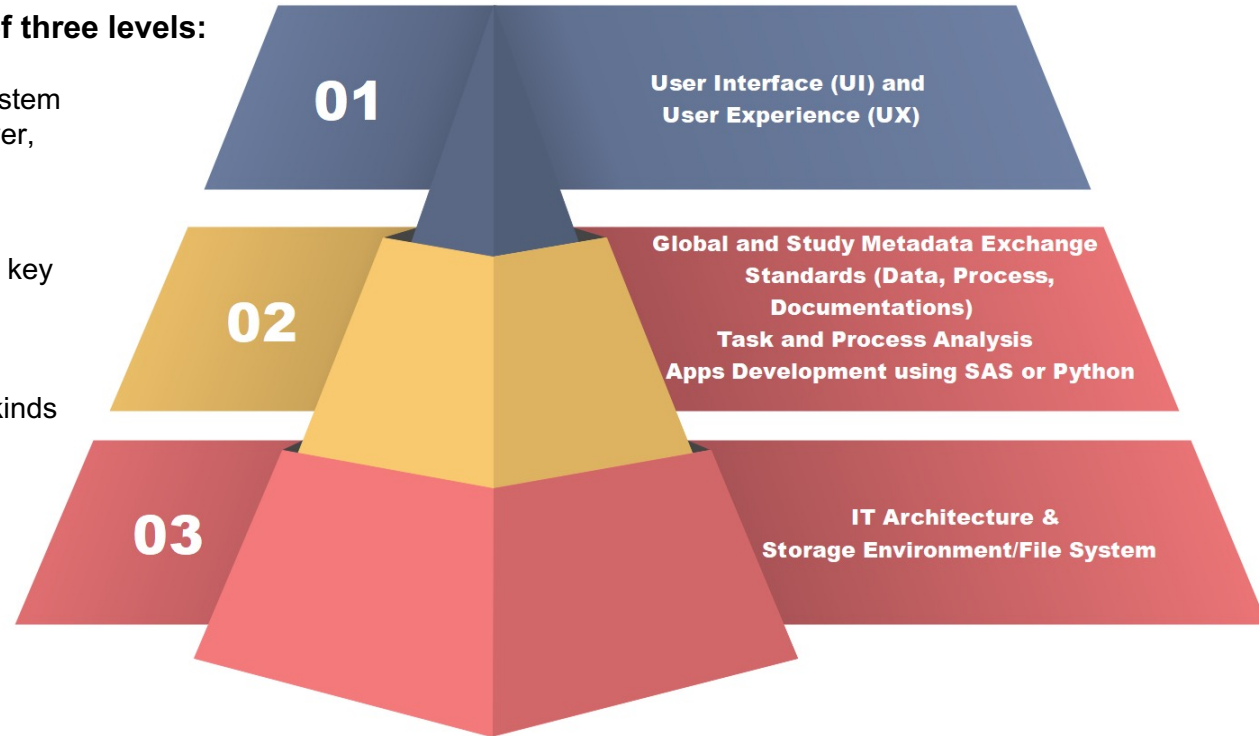
Automated Metadata-Exchange Process



Dizal-iSCP Structure

Dizal-iSCP is an ecosystem consisting of three levels:

- Lowest level is the foundation of whole system and includes IT architecture like SAS server, storage environment and file system
- Middle level contains various types of document including metadata file which is key to automation
- Highest level is the user interface through which users can get access to almost all kinds of software and files



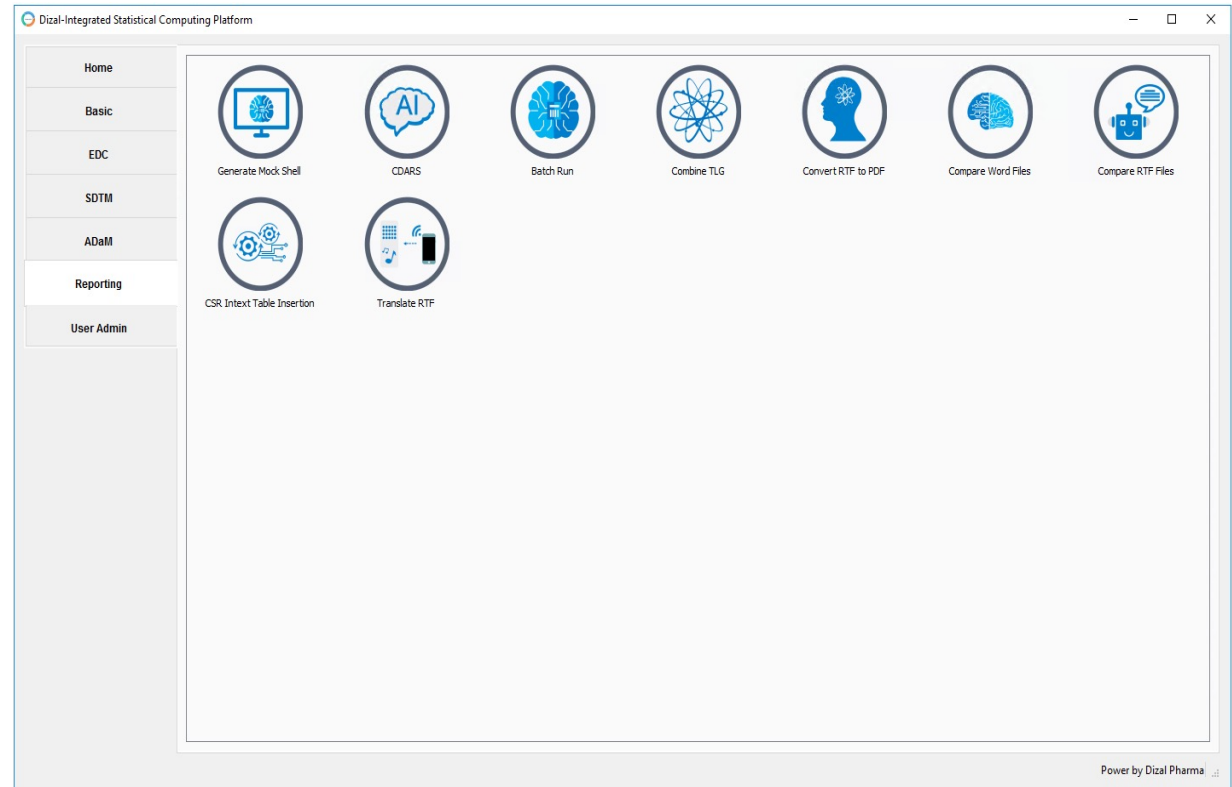
Programmers are not only the software users but also developers.

All the users can contribute to construction of Dizal-iSCP if they follow same development standards.

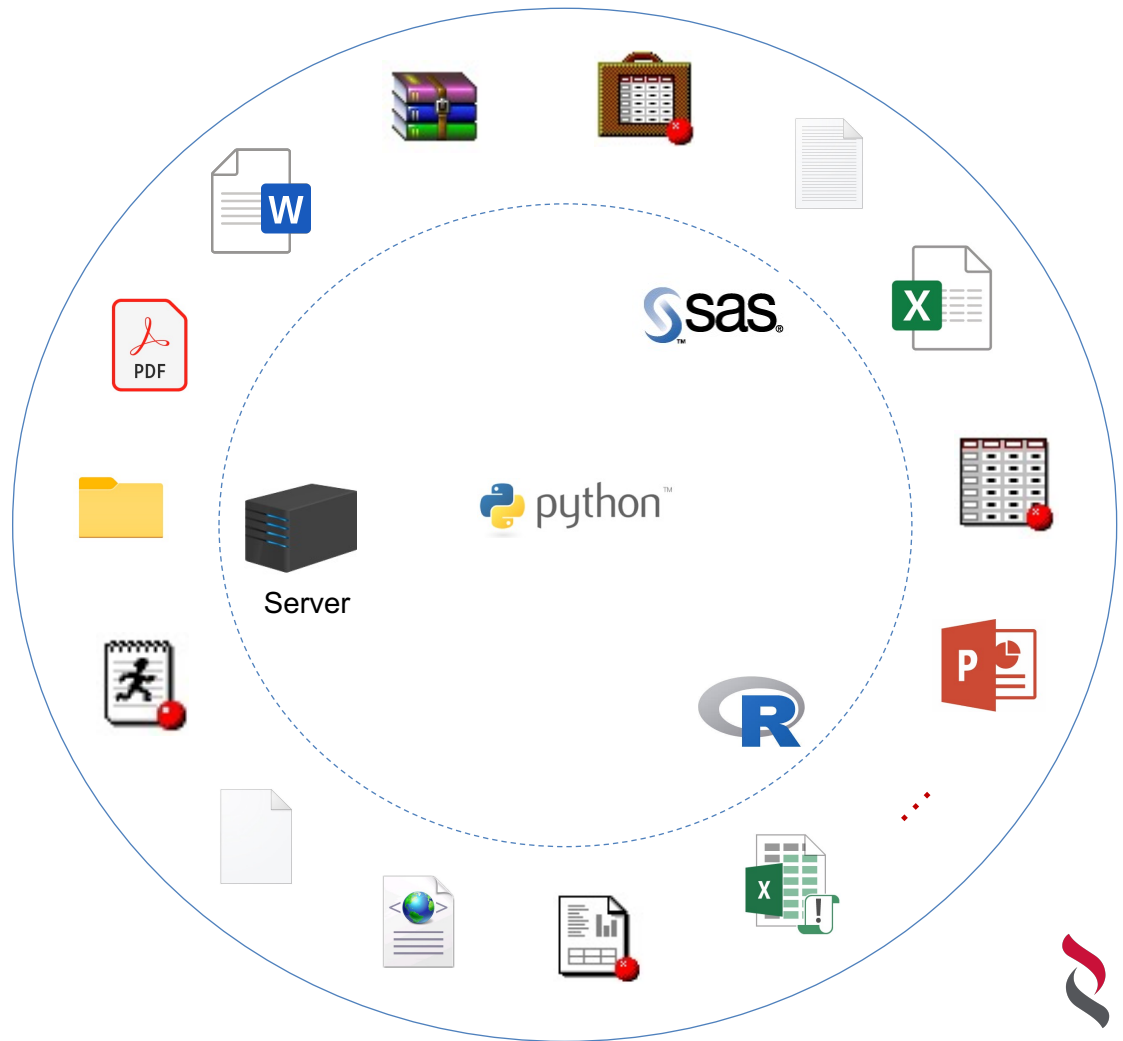


Dizal-iSCP User Interface

- Consists of Home tab, User Admin tab and task tabs
- Five task tabs are organized per analysis data flow
- Each task tab is made up of shortcut icons to standalone applications
- Users only need to click several buttons and many daily tasks can be done



- Main Software include Python, Linux SAS, R
- Linux SAS and R is responsible for data analysis and reporting
- Python interacts with other types of software to avoid toggle and reduce manual work
 - ❖ SAS program
 - ❖ SAS dataset
 - ❖ SAS log
 - ❖ SAS lst
 - ❖ Word
 - ❖ Excel
 - ❖ PDF
 - ❖ HTML
 - ❖ PPT
 - ❖ ...





- Checkpoints set up in the system force users to abide by SOPs and thus reduce compliance risk. If users accidentally trigger the checkpoints by interacting with wrong interface elements, warning message will be prompted to guide users in right direction.
- Generate submission-ready artifacts including SDTM, ADaM, TLG and other files
- Applying automation and AI to reduce manual work and improve efficiencies

Case 1: SDTM Auto Mapping

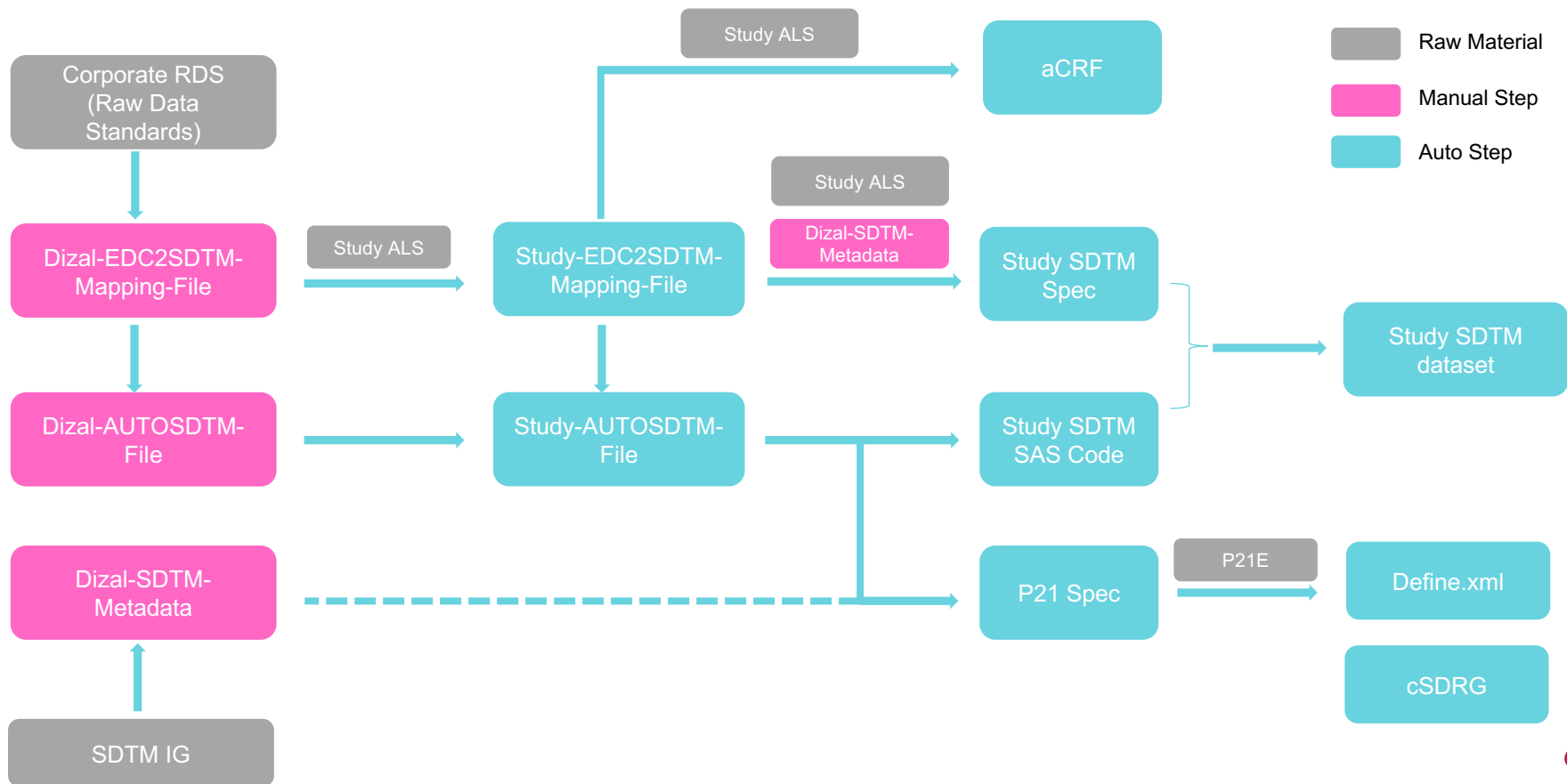
- Global Metadata and Process Analysis
- SDTM Auto Mapping Apps Development

Case 2: Automation of TLG

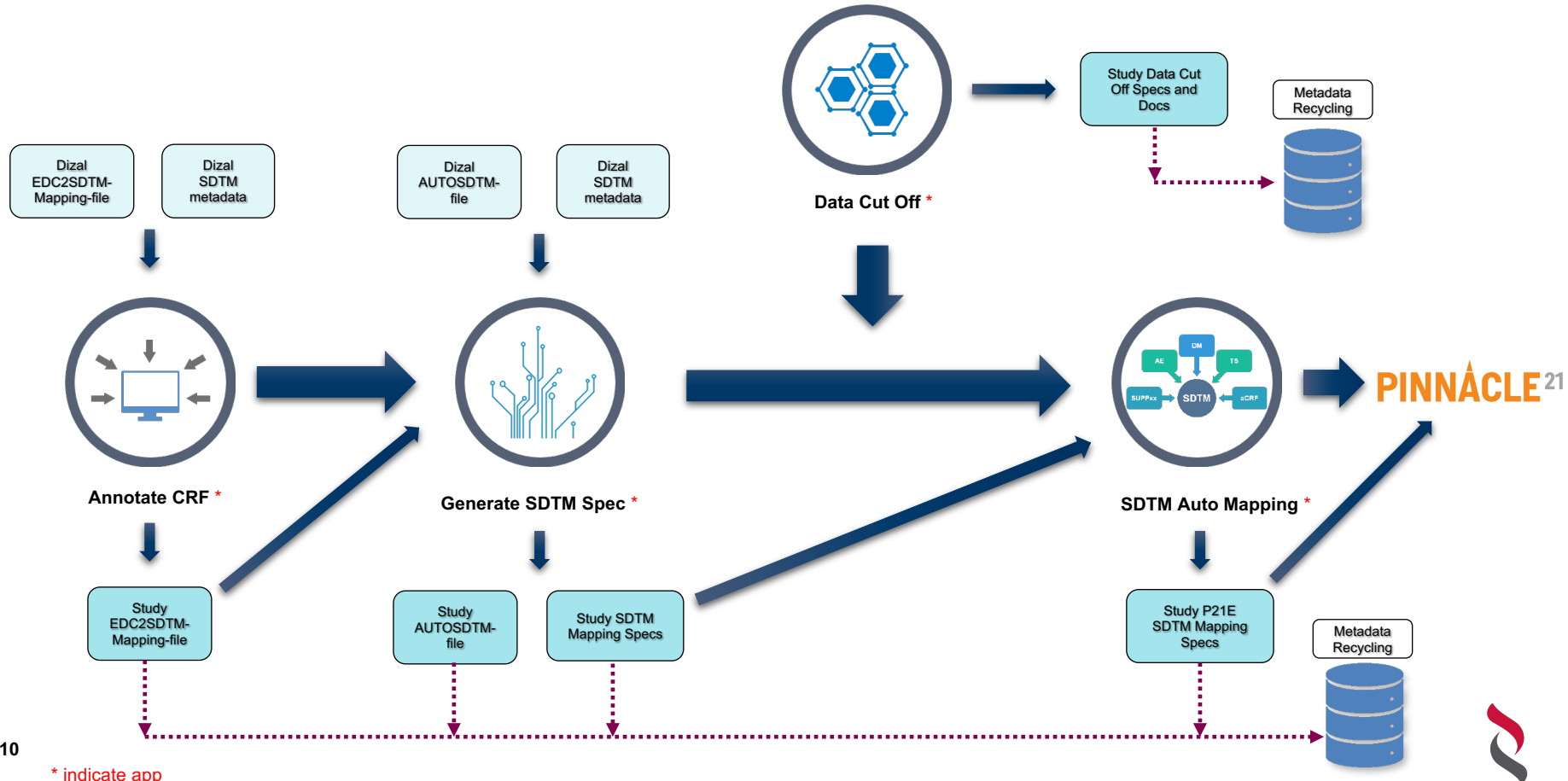
- Global Metadata and Process Analysis
- Table Shells and TLG Automation Apps Development



Case 1: SDTM Auto Mapping: Metadata and Process Analysis



SDTM Auto Mapping Apps Development



SDTM Auto Mapping Apps Development

Annotate CRF

ar-devar-trainingsstat-programmingstandard-process

Step 1: Select Required Input Documents

CRF file in pdf format:ar-trainings-stat-programming-standard-process-CRF.pdf

ALS file in xlsx format:ar-trainings-stat-programming-standard-process-ALS.xlsx

Previous edc2sdm mapping file if exists:

Generate edc2sdm template with global template

Generate edc2sdm template with old edc2sdm

Review PreText-Not-Extracted

Step 2: Fill edc2sdm mapping file

edc2sdm mapping file in xlsx format:edc2sdm-mapping-file-2021-08-31.xlsx

md-sdtm file in xlsx format:

Open and Edit

Verify Requirements

Review Check Result

Link edc2sdm and Position

Generate bookmark template

Step 3: Generate PDF Containing Bookmark

Bookmark file in xlsx format:crf-bookmark-2021-08-31.xlsx

Annotated CRF file in pdf format:aCRF-2021-08-31.pdf

Open and Edit (if needed)

Add Bookmark

Step 4: Generate FDF containing Annotation

File linking edc2sdm and Position:edc2sdm-position-for-fdf-2021-08-31.xlsx

Open and Edit (if needed)

Generate FDF

Step 5: Import FDF into CRF PDF

FDF file to be imported into CRF:edc2sdm-fdf-2021-08-31.fdf

Import FDF

Open aCRF and Edit

Check Annotation-Annotation Overlap

Open chk-anno-anno

Check Annotation-CRF_Text Overlap

Open chk-anno-crf

Open docs/sdtm Folder

Generate SDTM Spec

ar-devar-trainingsstat-programmingstandard-process

General Domain

TA

TE

TV

TD

TI

TS

Combine All Sheets

edc2sdm Map File:edc2sdm-mapping-file-2021-08-31.xlsx

ALS File:ar-trainings-stat-programming-standard-process-ALS.xlsx

SDTM Metadata:md-sdtm32.xlsx

SDTM Spec:SDTM-Spec-2021-08-31.xlsx

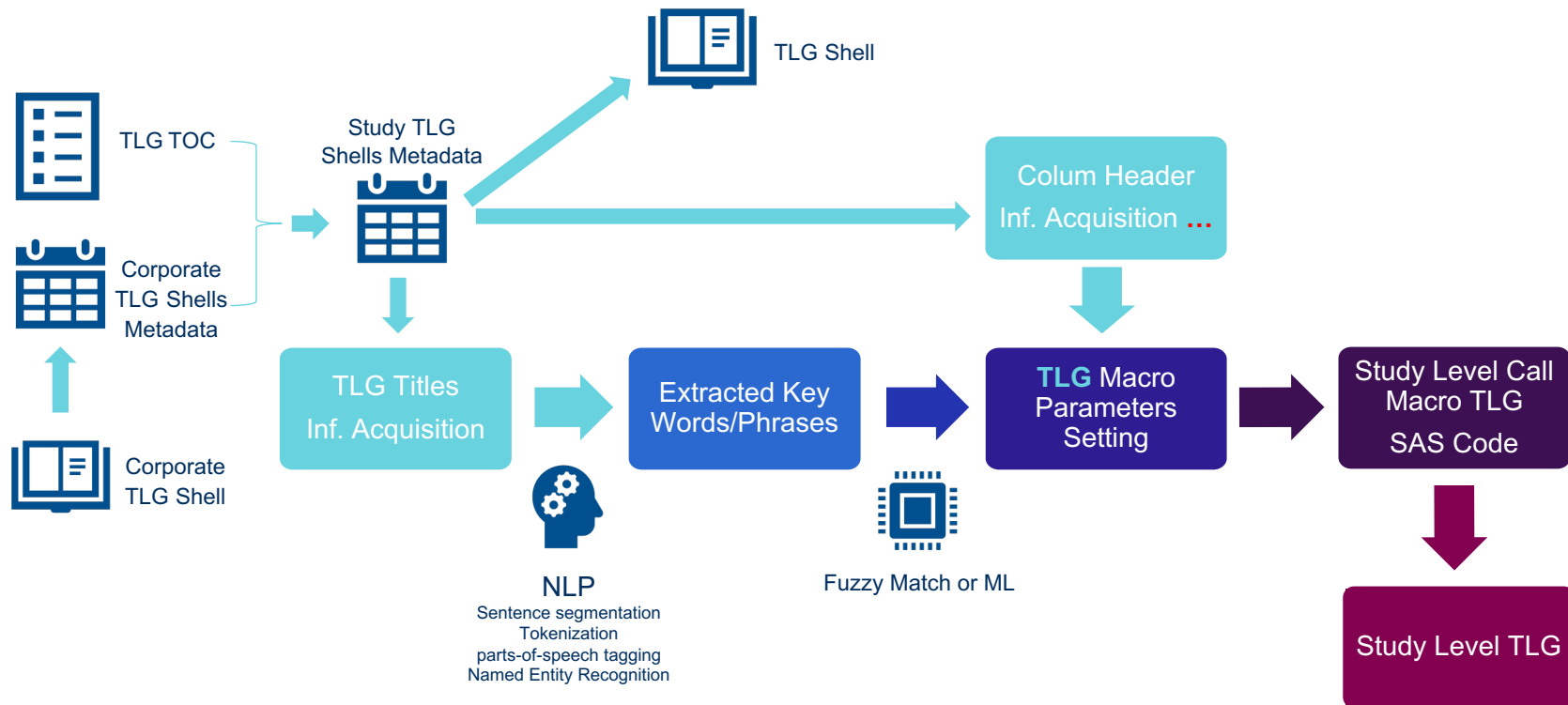
Language:en

Generate dataset and variables Sheet

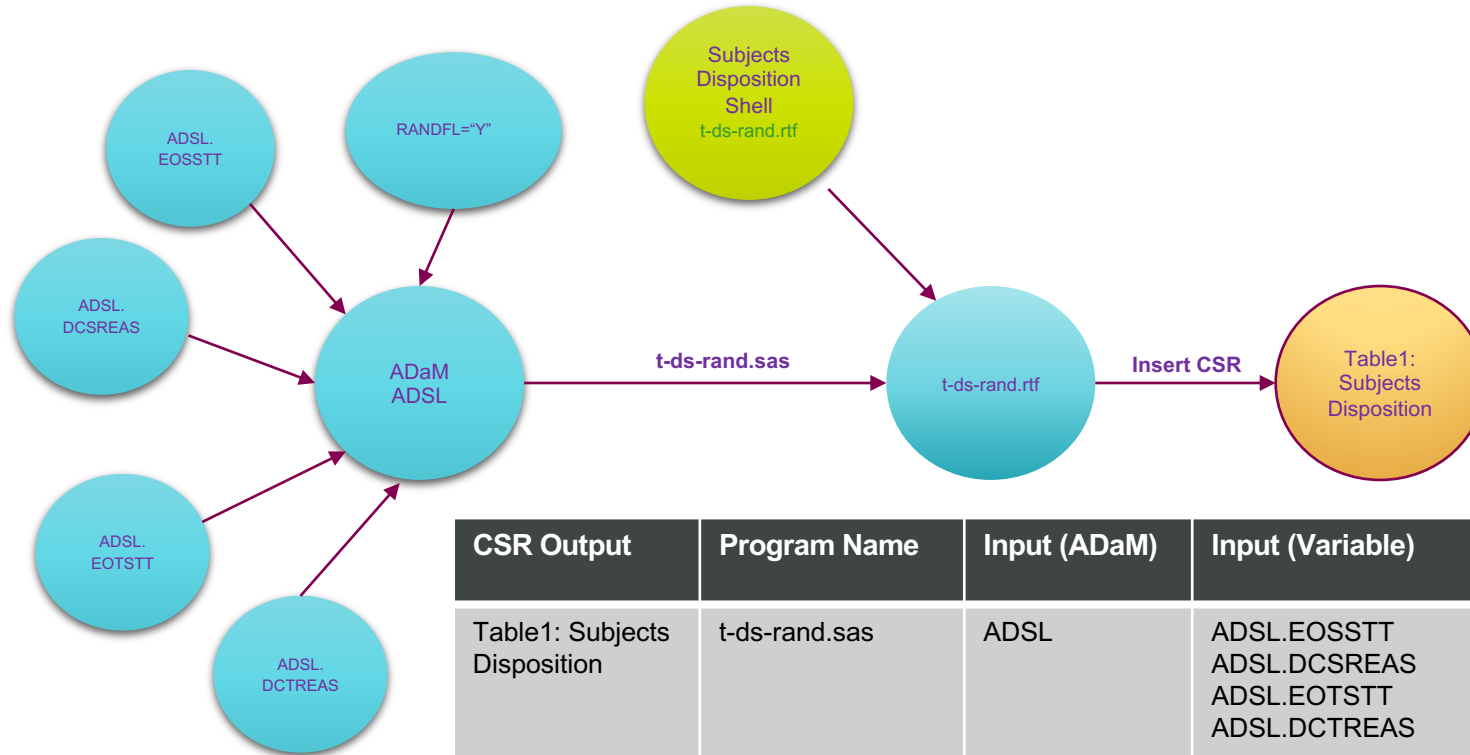
Open and Edit



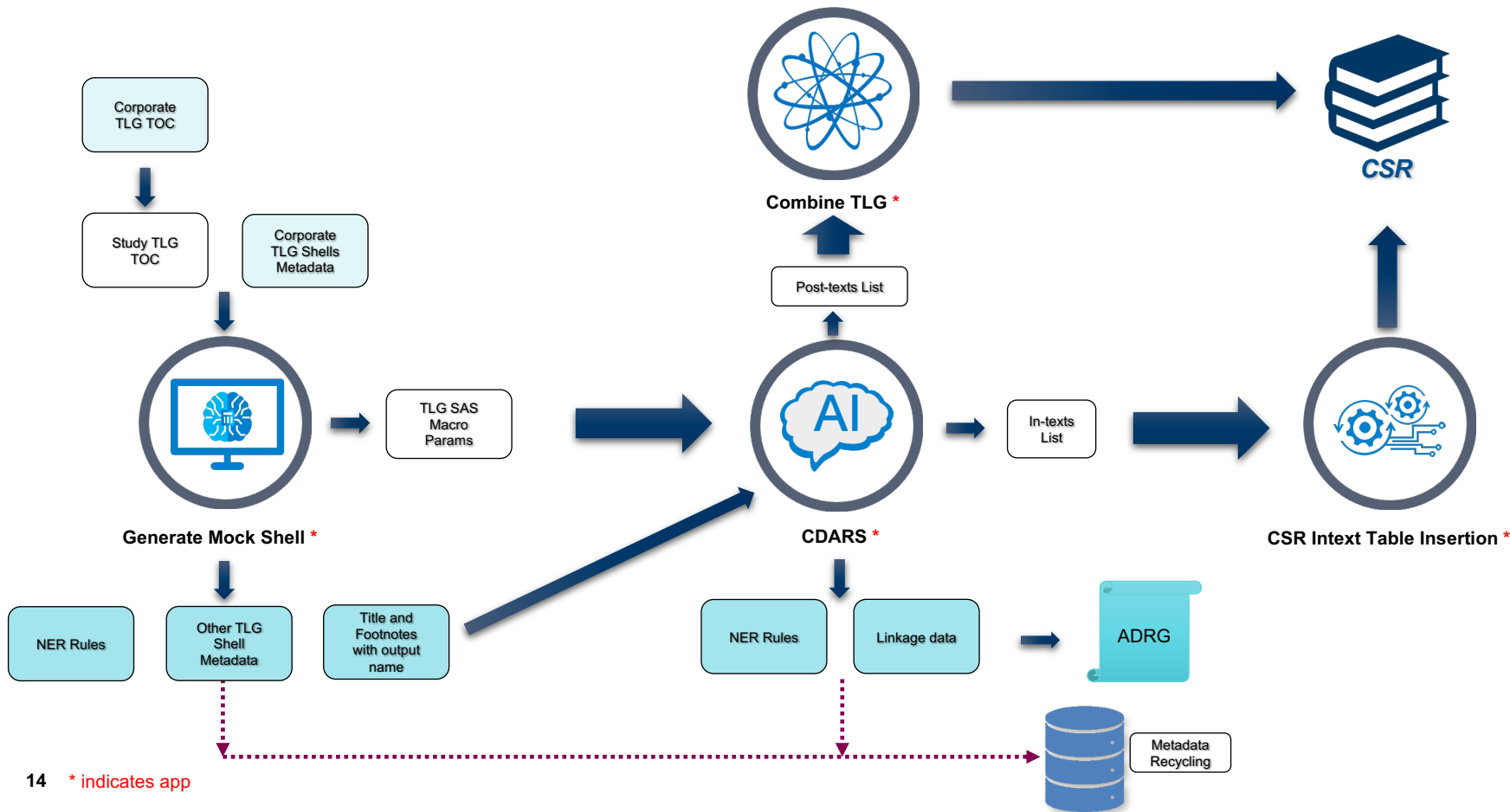
Case 2: Automation of TLG



How the “Subjects Disposition” Table Generated



Automation of TLG Apps Development



Automation of TLG Apps Development

Generate Mock Shell

ar-dev ar-trainings

Overall TOC

Step 1: Generate Outfile_Name

TOC Excel File ar-trainings-stat-programming-standard-process-toc.xlsx

Generate Outfile_Name Edit Outfile_Name

Step 2: Generate Overall XML File and Standalone XML File

Generate XML Generate/Combine XML

Step 3: Conversion between XML and Tree

Row XML ar-trainings-stat-programming-standard-process-row-xml.xml

Column XML ar-trainings-stat-programming-standard-process-column-xml.xml

Row xml2tree Column xml2tree

Row tree2xml Column tree2xml

Split Row XML Split Column XML

Step 4: Generate Initial Title and Footnote Excel from TOC

Title and Footnote Excel File ar-trainings-stat-programming-standard-process-title-footnote.xlsx

Generate Title and Footnote Excel Open Title and Footnote Excel

Step 5: Generate Shell from XML file

TFL Shell File ar-trainings-stat-programming-standard-process-TFL-Shell.rtf

Generate TFL Shell XML2RTF Combine RTF

View & Check Open TFL Shell Open System Folder

Open mock-shell-xml Open mock-shell-rtf

CDARS

ar-dev ar-trainings stat-programming standard-process

Generate All SAS Program

Language en

Destination A

Sub Folder reports

Title Footnote ar-trainings-stat-programming-standard-prc

xlsx2sas Open Title Footnote

output_pgm_path Y

Generate SAS File Open Program Folder

A and R Grid Log Open Scripts

Shell_ID	Output_Type	Output_Number	Title	Population	Outfile_Name	Dev_Programmer	Macro User Manual
TABLE	T	14.1.1.1a	Summary of ...	Safety Analys...	t-ds-pa	SP1	
DS0101	T	14.1.1.1b	Summary of ...	Safety Analys...	t-ds-pb	SP1	m-ar-ds-macro-user-manual
DS0101	T	14.1.1.1c	Summary of ...	Safety Analys...	t-ds-pc		m-ar-ds-macro-user-manual
DS0103	T	14.1.1.2a	Summary of ...	All Subjects	t-ds-pa		m-ar-scrmf-macro-user-manual
DS0102	T	14.1.2.1a	Summary of ...	Safety Analys...	t-ds-pa		m-ar-pop-macro-user-manual
DS0102	T	14.1.2.1b	Summary of ...	Safety Analys...	t-ds-pb		m-ar-pop-macro-user-manual
DS0102	T	14.1.2.1c	Summary of ...	Safety Analys...	t-ds-pc		m-ar-pop-macro-user-manual
DV0101	T	14.1.3.1a	Summary of ...	Safety Analys...	t-ds-pa		m-ar-dv-macro-user-manual
DV0101	T	14.1.3.1b	Summary of ...	Safety Analys...	t-ds-pb	SP2	m-ar-dv-macro-user-manual
DV0101	T	14.1.3.1c	Summary of ...	Safety Analys...	t-ds-pc	SP2	m-ar-dv-macro-user-manual
DM0101	T	14.1.4.1a	Summary of ...	Safety Analys...	t-dm-pa	SP1	m-ar-cat-cont-macro-user-manual
DM0101	T	14.1.4.1b	Summary of ...	Safety Analys...	t-dm-pb	SP1	m-ar-cat-cont-macro-user-manual
DM0101	T	14.1.4.1c	Summary of ...	Safety Analys...	t-dm-pc	SP1	m-ar-cat-cont-macro-user-manual
MH0101	T	14.1.4.2a	Summary of ...	Safety Analys...	t-mh-pa	SP2	m-ar-mh-medhis-macro-user...
MH0101	T	14.1.4.2b	Summary of ...	Safety Analys...	t-mh-pb	SP2	m-ar-mh-medhis-macro-user...
MH0101	T	14.1.4.2c	Summary of ...	Safety Analys...	t-mh-pc	SP2	m-ar-mh-medhis-macro-user...
CM0101	T	14.1.4.3a	Summary of Pri...	Safety Analys...	t-pr-cm-pa	SP2	m-ar-cm-conmed-macro-user...
CM0101	T	14.1.4.3b	Summary of Pri...	Safety Analys...	t-pr-cm-pb	SP2	m-ar-cm-conmed-macro-user...
CM0101	T	14.1.4.3c	Summary of Pri...	Safety Analys...	t-pr-cm-pc	SP2	m-ar-cm-conmed-macro-user...
CM0101	T	14.1.4.4a	Summary of ...	Safety Analys...	t-cm-pa	SP2	m-ar-cm-conmed-macro-user...
CM0101	T	14.1.4.4b	Summary of ...	Safety Analys...	t-cm-pb	SP2	m-ar-cm-conmed-macro-user...
CM0101	T	14.1.4.4c	Summary of ...	Safety Analys...	t-cm-pc	SP2	m-ar-cm-conmed-macro-user...
DH0101	T	14.1.4.5a	Summary of ...	Safety Analys...	t-dh-pa	SP2	
DH0101	T	14.1.4.5b	Summary of ...	Safety Analys...	t-dh-pb	SP2	
DH0101	T	14.1.4.5c	Summary of ...	Safety Analys...	t-dh-pc	SP2	
DT0101	T	14.1.4.6a	Summary of Pri...	Safety Analys...	t-prcatb-pa	SP2	
DT0101	T	14.1.4.6b	Summary of Pri...	Safety Analys...	t-prcatb-pb	SP2	



Summary

- A statistical programming platform can be established that the programming activities are integrated into an ecosystem of statistical programming based on the concept of IoT
- The Statistical Computing Platform should incorporate innovative system architecture and technical ideas, ensure high-standard quality, maximize analysis & reporting efficiency, and guarantee deliverable readiness
- Standards including data standards and tools are based on science, regulatory requirements, and industry best practices
- Design and develop the platform according to the company's business needs and do not blindly increase feature. But the platform should have the quality of scalability and extensibility



Acknowledgements

Thanks Weibin Cai, Jundong Ma, Jianyong Tong, Zongming Guo and Wei Zhang for the great supports



Contact Information

Author Name: Zhiping Yan

Company: Dizal

Email: zhiping.yan@dizalpharma.com

Author Name: Huadan Li

Company: Dizal

Email: huadan.li@dizalpharma.com







**THANK YOU FOR YOUR
PARTICIPATION**