Automation of SDTM Generation & Artifacts using CDISC 360 enriched standards
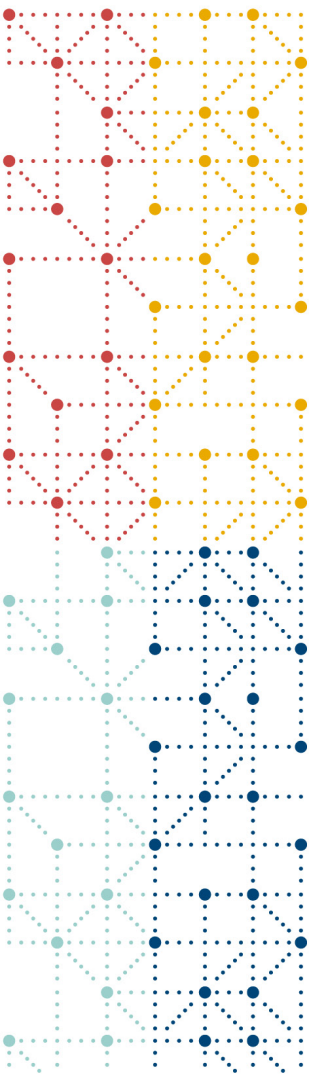
Bhavin Busa (Vita Data Sciences), Jianhui [Jimmy] Zhao (Allergan)
CDISC 360: The Journey So Far and the Road Ahead
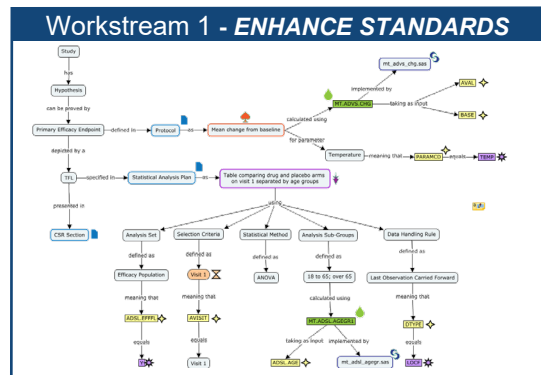*April 28, 2020*

cdisc

Agenda

1. Workstream 6 Introduction
2. Current State: CDASH to SDTM Execution
3. CDISC 360 Enriched Metadata
4. Future State with Concept-based Standards : CDASH to SDTM Execution
5. Process Flow for CDISC 360 Proof of Concept
6. Machine-readable Mapping Specifications
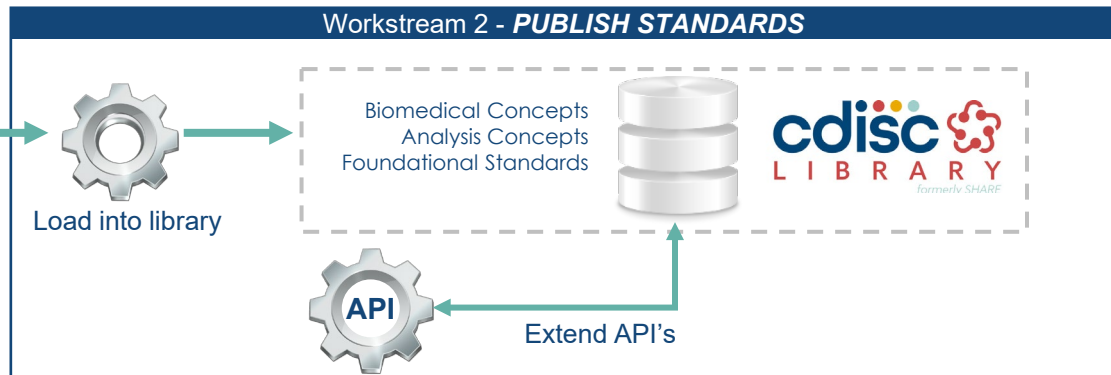7. Learnings so far
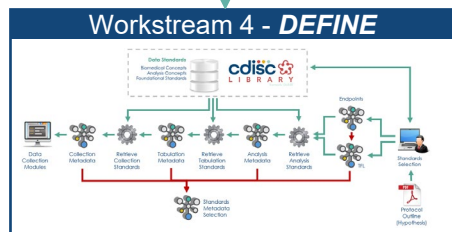
Workstream 6 Introduction

CDISC 360 Workstreams

Workstream 1 - *ENHANCE STANDARDS*

Create concepts in knowledge graphs

Load into library

Workstream 2 - *PUBLISH STANDARDS*

Biomedical Concepts
Analysis Concepts
Foundational Standards

cdisc LIBRARY
formerly SHARP

API

Extend API's

Transform concepts in machine readable form

Study Metadata Library

Workstream 4 - *DEFINE*

Identify and select standards specification (Use Case 1)

Workstream 5 - *BUILD*
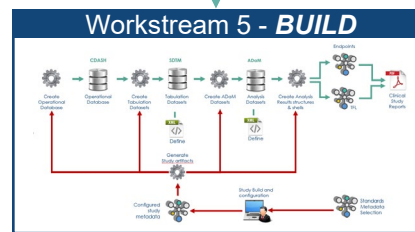
Configure study specification and create artifacts (Use Case 2)

Workstream 6 - *EXECUTE*
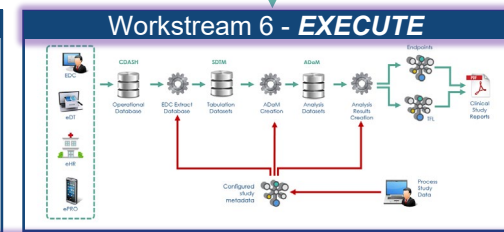
Automatically process and transform data (Use Case 3)

Use Case 3 (Workstream 6): Execute
Automatic population of data into artifacts
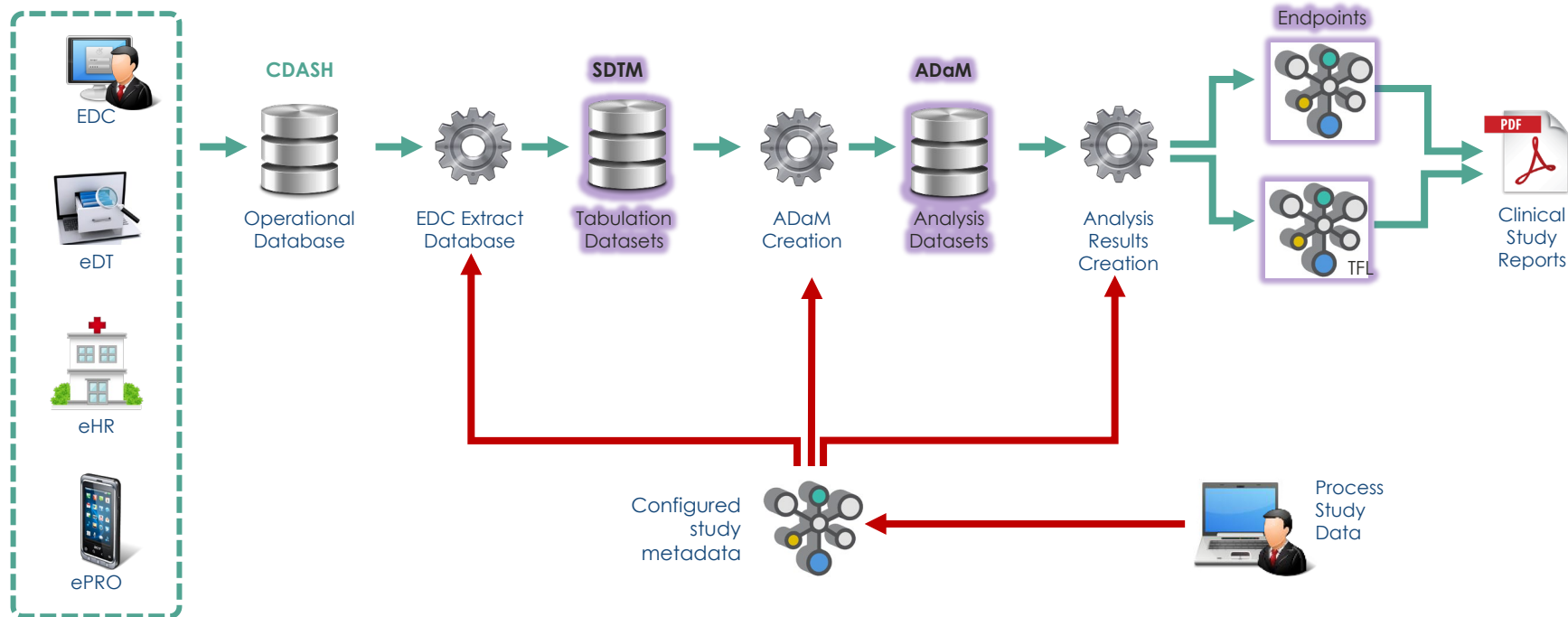
Workstream 6 & Task Team Leads

Workstream 6 Lead

Bhavin Busa,
Vita Data Sciences

SDTM/ADaM Automation Task Team Leads

TFL Automation Task Team Leads

Kaja Najumudeen,
TalentMine

Jianhui Zhao,
Allergan
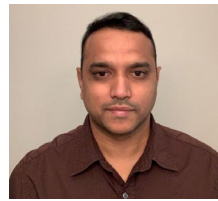
Prasanna Murugesan,
AstraZeneca

Stuart Malcolm,
Frontier Science

cdisc

Current State - without Concept-based Standards

Current State - without Concept-based Standards: CDASH to SDTM Execution

Foundational Standards

cdisc LIBRARY formerly SHARE

CRF* → SDTM Specs* → SAS Programs[+] → SDTM Datasets[+] → SDTM Define & aCRF[+] → Submission[+] XPT files, Define, aCRF, SDRG

Specify — *Build/Execute* — *Report*
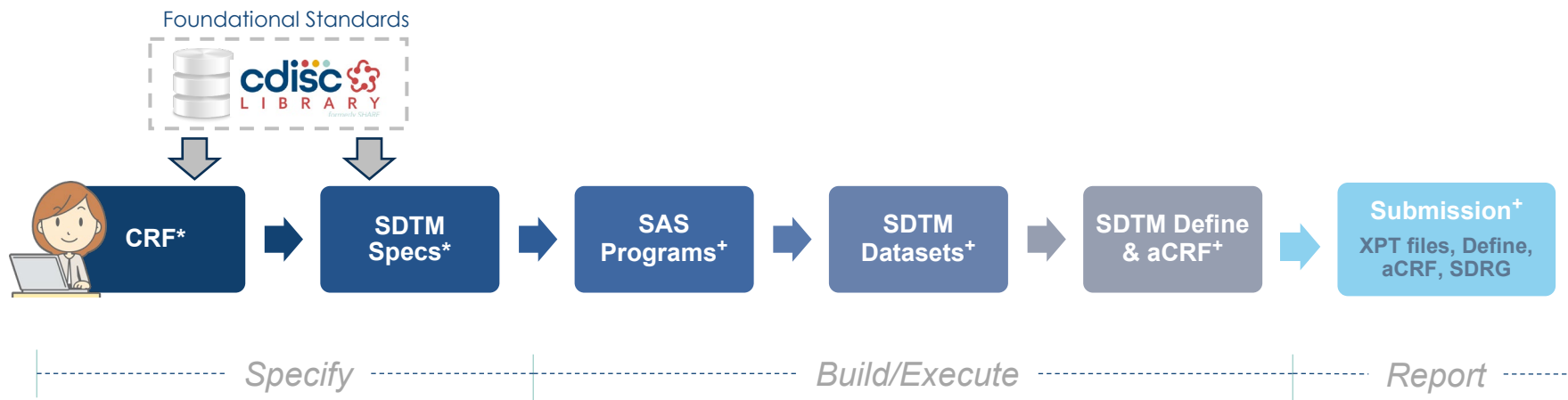
* Manual Process
+ Manual or Semi-automated Execution

cdisc

9

CDISC 360 Enriched Metadata

Machine-readable CDISC 360 Enriched Metadata

Concept-based Standards: Biomedical Concept

- Triple Store

- Linking controlled terminology to the variable - standardize value level metadata

- Linked derivations and algorithms to variable(s)

- Include process metadata (ETL instructions)

- Machine readable definition of validation rules
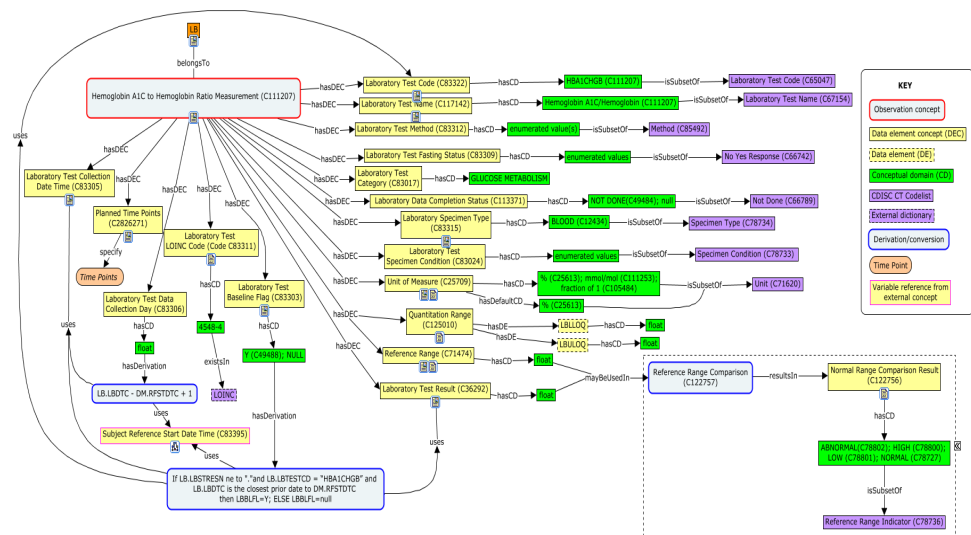
Linked Graph Model: Importing Concept-based Standards
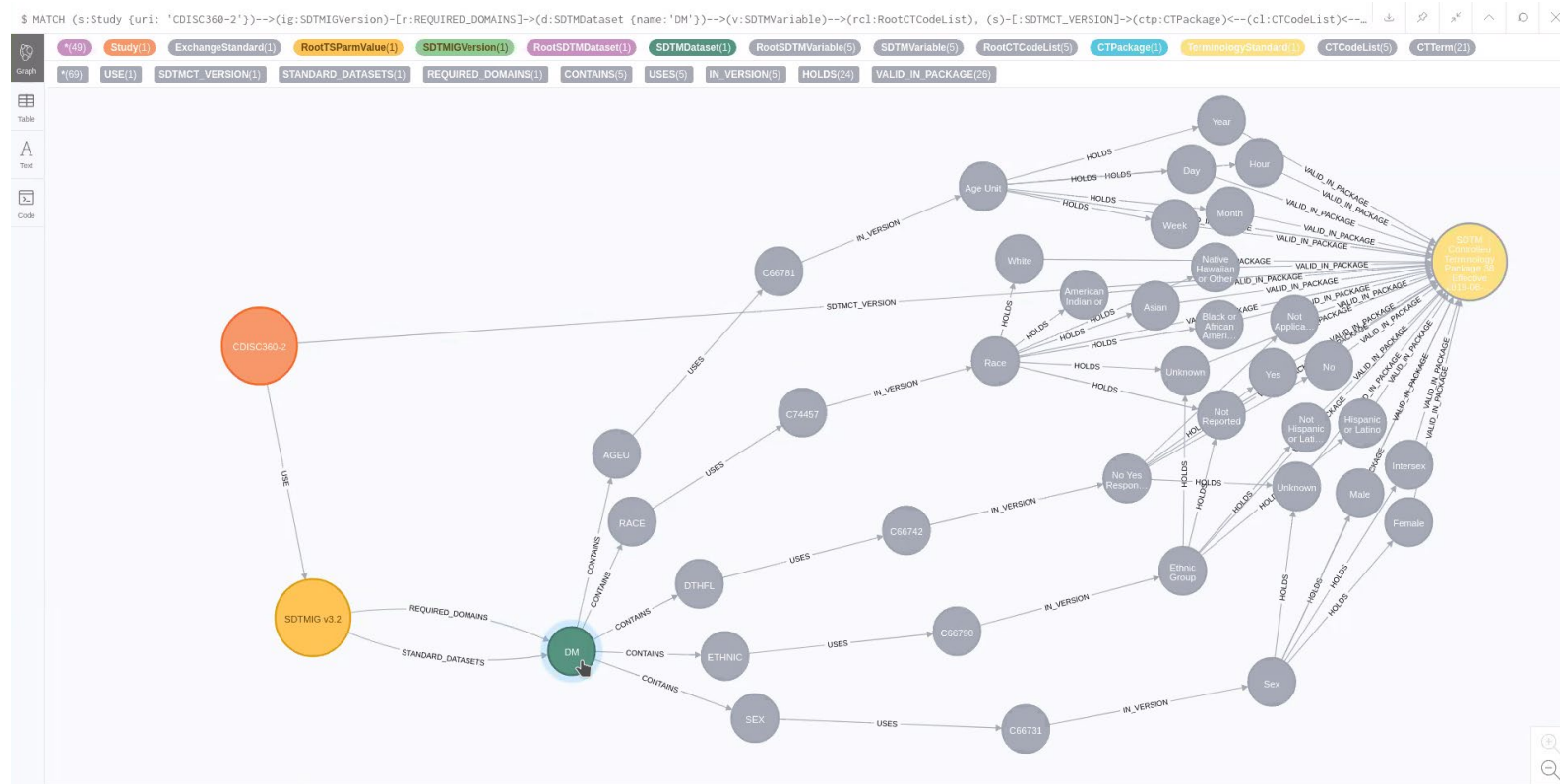
Reference: 'Study Design and Configuration using CDISC 360 concept-based standards', *Mikkel Traun,* Nicolas de Saint Jorre, *28th April 2020*

Study Build of ODM.XML and Define.XML

ODM CRF Generated using Biomedical Concepts, Bindings, & Standards

ODM-based Vital Signs (VS) CRF

```xml
<MetaDataVersion Description="CDASH BC CRF Example" Name="CDASH CRF Example" OID="MDV.CDISC360.DEMO1">
    <FormDef Name="VS Form" OID="F.VS" Repeating="Yes">
        <ItemGroupRef ItemGroupOID="IG.BC.VS.COMMON" Mandatory="No"/>
        <ItemGroupRef ItemGroupOID="IG.BC.VS.TEMPERATURE" Mandatory="No"/>
        <ItemGroupRef ItemGroupOID="IG.BC.VS.HEIGHT" Mandatory="No"/>
        <ItemGroupRef ItemGroupOID="IG.BC.VS.DIASTOLICBP" Mandatory="No"/>
        <ItemGroupRef ItemGroupOID="IG.BC.VS.SYSTOLICBP" Mandatory="No"/>
        <ItemGroupRef ItemGroupOID="IG.BC.VS.WEIGHT" Mandatory="No"/>
        <ItemGroupRef ItemGroupOID="IG.BC.VS.HEARTRATE" Mandatory="No"/>
    </FormDef>
    <ItemGroupDef Name="VS Common" OID="IG.BC.VS.COMMON" Repeating="No">
        <ItemRef ItemOID="IT.BC.VS.STUDYID" Mandatory="Yes" OrderNumber="1"/>
        <ItemRef ItemOID="IT.BC.VS.SITEID" Mandatory="Yes" OrderNumber="2"/>
        <ItemRef ItemOID="IT.BC.VS.SUBJID" Mandatory="Yes" OrderNumber="3"/>
        <ItemRef ItemOID="IT.BC.VS.VISIT" Mandatory="No" OrderNumber="4"/>
        <ItemRef ItemOID="IT.BC.VS.VSPERF" Mandatory="No" OrderNumber="6"/>
        <ItemRef ItemOID="IT.BC.VS.VSDAT" Mandatory="No" OrderNumber="7"/>
        <ItemRef ItemOID="IT.BC.VS.VSTIM" Mandatory="No" OrderNumber="8"/>
    </ItemGroupDef>
    <ItemGroupDef Name="VS Temperature" OID="IG.BC.VS.TEMPERATURE" Repeating="No">
        <ItemRef ItemOID="IT.BC.VS.TEMPERATURE.VSTEST" Mandatory="Yes" OrderNumber="1"/>
        <ItemRef ItemOID="IT.BC.VS.TEMPERATURE.VSORRES" Mandatory="Yes" OrderNumber="2"/>
        <ItemRef ItemOID="IT.BC.VS.TEMPERATURE.VSORRESU" Mandatory="No" OrderNumber="3"/>
    </ItemGroupDef>
```

Stylesheet rendering of ODM VS CRF

Reference: 'CDISC Library: Integrating and Surfacing 360 Content', Sam Hume, October 16, 2019

Future State - with Concept-based Standards

Future State - with Concept-based Standards: CDASH to SDTM Execution
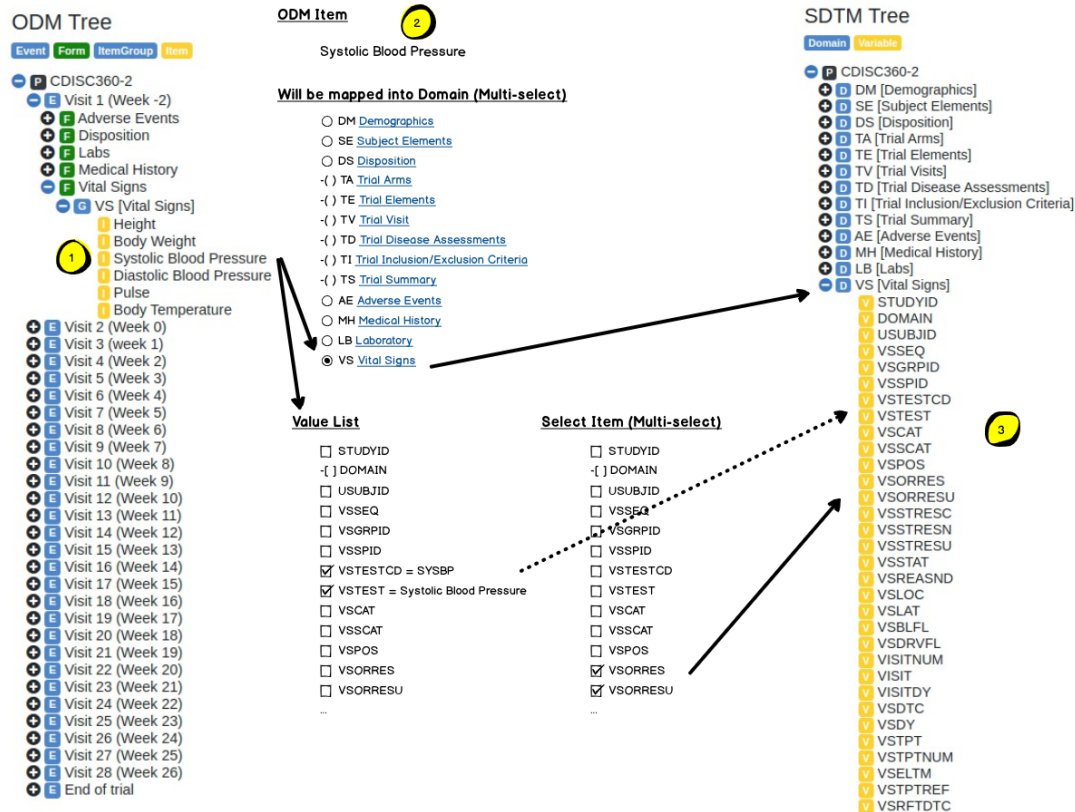

Biomedical Concepts
Foundational Standards

cdisc LIBRARY

Design

Study and Data Collection Designers

| ODM CRF | → | Define XML | → | SDTM Specs & aCRF | → | SDTM Programs | → | SDTM Datasets | → | Submission XPT files, Define, aCRF, SDRG |

Specify ---- *Build/Execute* ---- *Report*

= Automated Process
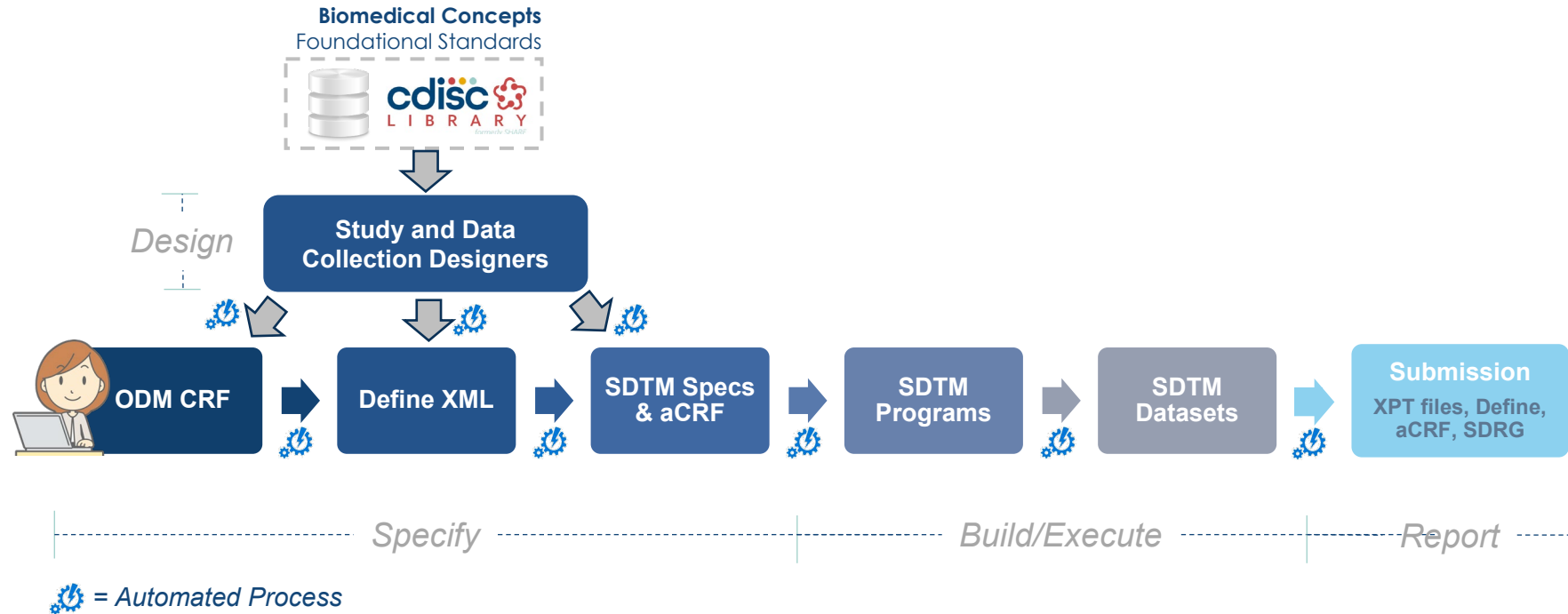
Process Flow for CDISC 360 Proof of Concept (PoC)

Process Flow for CDISC 360 PoC
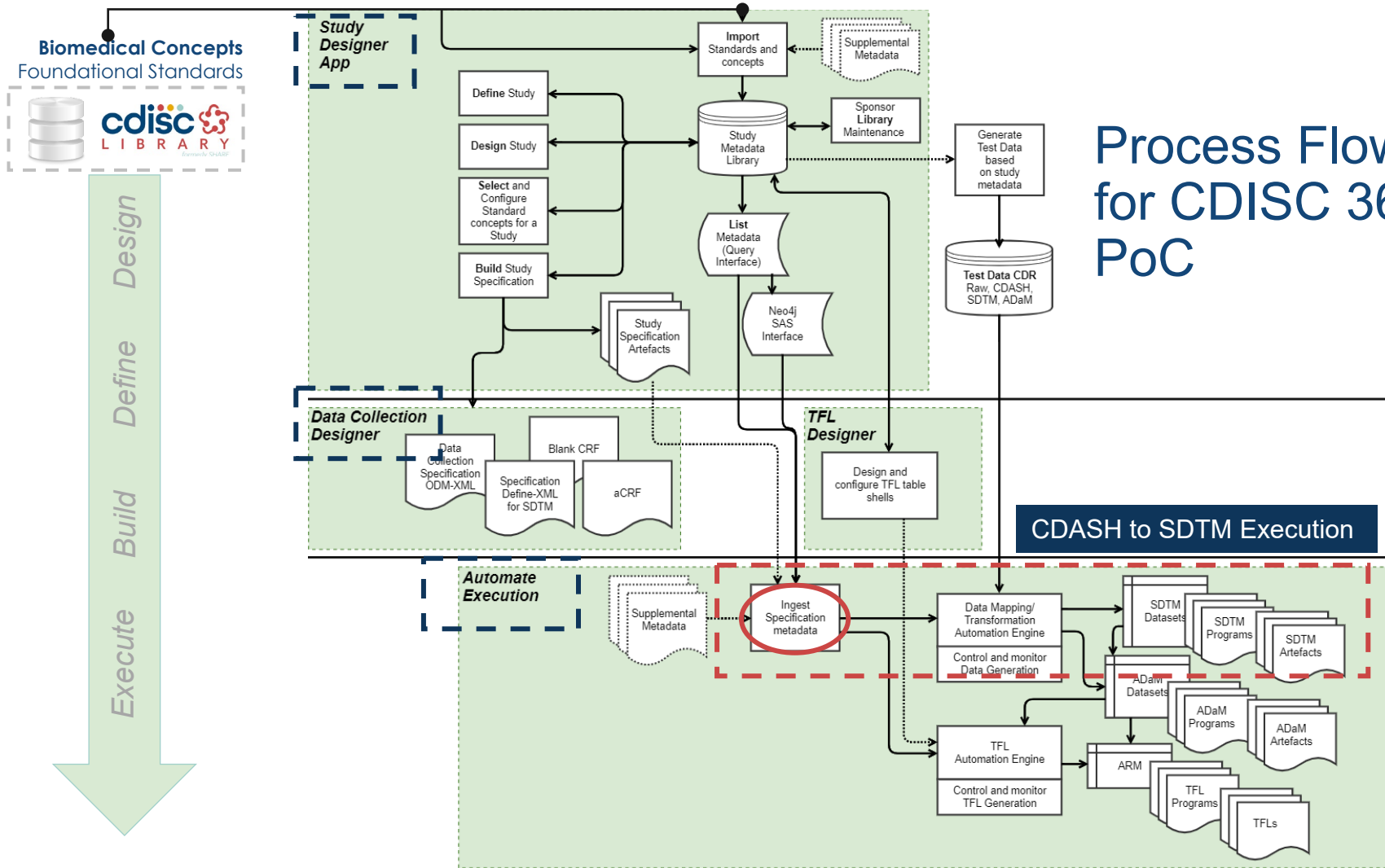
Machine-readable Mapping Specifications

Essential Elements for Machine-readable Mapping Specifications

We break down the essential elements in 2 dimensions to meet the 4 key aspects of the machine readability

Dimension 1

- <u>Source</u>: location (library name), datasets, processing sequence

- <u>Mapping</u>: fields needed to describe how source transits to target

- <u>Target</u>: location (library name), datasets, processing sequence, attributes (label, class, structure, purpose, etc.)

Dimension 2

- <u>Dataset Level</u>: Transit datasets from source to target

- <u>Variable Level</u>: Map variables from source to target

- <u>Value Level</u>: Map variables from source to target under different conditions

cdisc

Mapping Specifications: **Dimension 1**

	Source				Mapping					Target						

	A	B	C	D	F	G	H	I	J	K	L	M	N	O	P	Q
	Source Sequence	Source Library	Source Dataset	Source Variable	Map Sequence	Origin	Method	Comment	Code List	Target Library	Target Dataset	Target Variable	Target Description	Target Data Type	Target Legnth	Target Sorting Order
	1	CDASH	VS			Assigned		CDISC360-2		SDTM	VS	STUDYID	Study Identifier	text	10	1
	1	CDASH	VS			Assigned		VS	DOMAIN	SDTM	VS	DOMAIN	Domain Abbreviation	text	2	2
	1	CDASH	VS	SUBJID		Assigned	ALL.USUBJID			SDTM	VS	USUBJID	Unique Subject Identifier	text	14	3
	1	CDASH	VS			Assigned	VS.VSSPID			SDTM	VS	VSSPID	Sponsor-Defined Identifier	text	4	5
	1	CDASH	VS	VISIT		Convert			VISITNUM	SDTM	VS	VISITNUM	Visit Number	integer	8	16
	1	CDASH	VS	VISIT		Predecessor			VISIT	SDTM	VS	VISIT	Visit Name	text	18	17
	1	CDASH	VS	VSDAT		Assigned	VS.VSDTC			SDTM	VS	VSDTC	Date/Time of Measurements	date	10	19
	1	CDASH	VS	VISDAT		Assigned	VS.VSDTC			SDTM	VS	VSDTC	Date/Time of Measurements	date	10	19
	1	CDASH	VS			Derived	VS.VSBLFL			SDTM	VS	VSBLFL	Baseline Flag	text	1	14
	2	SDTM	DM	RFSTDTC		Derived	VS.VSDY			SDTM	VS	VSDY	Study Day of Vital Signs	integer	8	20
	2			VSDTC		Derived	VS.VSDY			SDTM	VS	VSDY	Study Day of Vital Signs	integer	8	20
	3	SDTM	SV	VISITDY	1	Predecessor				SDTM	VS	VISITDY	Planned Study Day of Visit	integer	8	18
	3	SDTM	SV	EPOCH	2	Predecessor			EPOCH	SDTM	VS	EPOCH	Epoch	text	9	15
	4				3	Assigned	VS.VSTESTCD		VSTESTCD	SDTM	VS	VSTESTCD	Vital Signs Test Short Name	text	6	6
	4				4	Derived	VS.VSORRES			SDTM	VS	VSORRES	Result or Finding in Original Units	text	4	9
	4				5	Derived	VS.VSORRESU		VSUNIT	SDTM	VS	VSORRESU	Original Units	text	9	10
	4				6	Assigned	VS.VSSTRESU		VSUNIT	SDTM	VS	VSSTRESU	Standard Units	text	9	13
	4				7	Derived	VS.VSSTRESN			SDTM	VS	VSSTRESN	Numeric Result/Finding in Standard Units	float	8	12
	4				8	Derived	VS.VSSTRESC			SDTM	VS	VSSTRESC	Character Result/Finding in Std Format	text	4	11
	4				9	Assigned	VS.VSPOS		VSPOS	SDTM	VS	VSPOS	Position	text	7	13
	5			VSTESTCD		Convert			VSTEST	SDTM	VS	VSTEST	Vital Signs Test Name	text	24	7
	5			VSTESTCD		Convert			VSCAT	SDTM	VS	VSCAT	Category for Vital Signs	text	16	8
	5					Derived	VS.VSSEQ			SDTM	VS	VSSEQ	Sequence Number	integer	8	4

cdisc

Mapping Specifications: **Dimension 2**

| | Source | | Mapping | | | | | Target | | |

Dataset Level

Source Sequence	Source Library	Source Dataset	Subset Condition	Pre Processing	Join Type	Join Timing	Merge Key	Target Sequence	Target Library	Target Dataset
1	CDASH	VS						5	SDTM	VS
2	SDTM	DM			TARGET	PRE	USUBJID	5	SDTM	VS
3	SDTM	SV			TARGET	PRE	USUBJID, VISITNUM	5	SDTM	VS
4					SORT		USUBJID, VISITNUM, VSDTC	5	SDTM	VS
5					SORT		USUBJID, VSTESTCDCD, VISITNUM, VSDTC	5	SDTM	VS

Variable Level

Source Sequence	Source Library	Source Dataset	Source Variable	Map Sequence	Origin	Method	Comment	Code List	Target Library	Target Dataset	Target Variable	Target Description	Target Data Type	Target Length	Target Sorting Order
4				3	Assigned	VS.VSTESTCD		VSTESTCD	SDTM	VS	VSTESTCD	Vital Signs Test Short Name	text	6	6
4				4	Derived	VS.VSORRES			SDTM	VS	VSORRES	Result or Finding in Original Units	text	4	9
4				5	Derived	VS.VSORRESU		VSUNIT	SDTM	VS	VSORRESU	Original Units	text	9	10
4				6	Assigned	VS.VSSTRESU		VSUNIT	SDTM	VS	VSSTRESU	Standard Units	text	9	13
4				7	Derived	VS.VSSTRESN			SDTM	VS	VSSTRESN	Numeric Result/Finding in Standard Units	float	8	12
4				8	Derived	VS.VSSTRESC			SDTM	VS	VSSTRESC	Character Result/Finding in Std Format	text	4	11
4				9	Assigned	VS.VSPOS		VSPOS	SDTM	VS	VSPOS	Position	text	7	13

Value Level

Source Sequence	Source Library	Source Dataset	Source Variable	Where Clause	Condition	Output	Map Sequence	Origin	Method	Comment	Code List	Target Library	Target Dataset	Target Variable	Target Data Type	Target Length	Significant Digits
3	WORK		VSORRES	VS.VSTESTCD.EQ.DIABP	DIABP_VSPERF = 'Y'	Y	7	Convert		best.		SDTM	VS	VSSTRESN	float	8	0
3	CDASH	VS	SYSBP_VSORRES	VS.VSTESTCD.EQ.SYSBP	SYSBP_VSPERF = 'Y'	Y	7	Convert		best.		SDTM	VS	VSSTRESN	float	8	0
3	CDASH	VS	HR_VSORRES	VS.VSTESTCD.EQ.PULSE	HR_VSPERF = 'Y'	Y	7	Convert		best.		SDTM	VS	VSSTRESN	float	8	0
3	CDASH	VS	TEMP_VSORRES	VS.VSTESTCD.EQ.TEMP	TEMP_VSPERF = 'Y'	Y	7	Convert		best.		SDTM	VS	VSSTRESN	float	8	1
3	CDASH	VS	HEIGHT_VSORRES	VS.VSTESTCD.EQ.HEIGHT	HEIGHT_VSPERF = 'Y'	Y	7	Derived	VS.VSSTRESN.item1			SDTM	VS	VSSTRESN	float	8	2
3	CDASH	VS	WEIGHT_VSORRES	VS.VSTESTCD.EQ.WEIGHT	WEIGHT_VSPERF = 'Y'	Y	7	Convert		best.		SDTM	VS	VSSTRESN	float	8	0

cdisc

Mapping Specifications: **Dataset Level**

Source Sequence	Source Library	Source Dataset	Subset Condition	Pre Processing	Join Type	Join Timing	Merge Key	Target Sequence	Target Library	Target Dataset
1	CDASH	VS						5	SDTM	VS
2	SDTM	DM			TARGET	PRE	USUBJID	5	SDTM	VS
3	SDTM	SV			TARGET	PRE	USUBJID, VISITNUM	5	SDTM	VS
4					SORT		USUBJID, VISITNUM, VSDTC	5	SDTM	VS
5					SORT		USUBJID, VSTESTCDCD, VISITNUM, VSDTC	5	SDTM	VS

❶ ❷ ❸ ❹ ❺

```
data VS1;
    set CDASH.VS;
    /********
     variable level: Source Sequence = 1
     ********/
run;
```
❶

```
proc sort data=VS1;  by SUBJID;
proc sort data=CDAHS.DM OUT=DM2; by USUBJID;

data VS2;
    merge DM2(in=a) VS1(in=b);
    by USUBJID;
    if b;
    /********
     variable level: Source Sequence = 2
     ********/
run;
```
❷

```
... Sequence 3, 4
```
❸ ❹

```
proc sort data=VS4;
    by USUBJID VSTESTCD VISITNUM VSDTC;
run;

data SDTM.VS;
    set VS4;
    by USUBJID VSTESTCD VISITNUM VSDTC;

 /********
    variable level: Source Sequence = 6
    ********/

run;
```
❺

cdisc

Mapping Specifications: **Variable Level**

Source Sequence	Source Library	Source Dataset	Source Variable	Map Sequence	Origin	Method	Comment	Code List	Target Library	Target Dataset	Target Variable	Target Description	Target Data Type	Target Legnth	Target Sorting Order
1	CDASH	VS			Assigned		VS	DOMAIN	SDTM	VS ❶	DOMAIN	Domain Abbreviation	text	2	2
1	CDASH	VS	SUBJID		Assigned	ALL.USUBJID			SDTM	VS ❷	USUBJID	Unique Subject Identifier	text	14	3
1	CDASH	VS	VISIT		Convert			VISITNUM	SDTM	VS ❸	VISITNUM	Visit Number	integer	8	16
1	CDASH	VS	VISIT		Predecessor			VISIT	SDTM	VS ❹	VISIT	Visit Name	text	18	17
1	CDASH	VS	VSDAT		Assigned	VS.VSDTC			SDTM	VS ❺	VSDTC	Date/Time of Measurements	date	10	19
1	CDASH	VS			Derived	VS.VSBLFL			SDTM	VS ❻	VSBLFL	Baseline Flag	text	1	14

ID	Description	Function	Parameter
❷ ALL.USUBJID	Concatenation of STUDYID and SUBJID	Concatenate	dot/STUDYID/SUBJID
❺ VS.VSDTC	Convert assessment date (VISDAT/VSDAT) to ISO8601 date format.	ISODTC	VISDAT/VSDAT
❻ VS.VSBLFL	Baseline flag set to Y when the assessment is collected at the visit marked as baseline in the trial flowchart.	Baseline	"VISIT/VISIT 2 (WEEK 0)"

SAS Code

❷
```
USUBJID = catx('.', STUDYID, SUBJID);
```
❺
```
if not missing(VISDAT) then
    VSDTC = put(VISDAT, e8601da.);
else if not missing(VSDAT) then
    VSDTC = put(VSDAT, e8601da.);
```
❻
```
if VISIT = "VISIT 2 (WEEK 0)" then VSBLFL = 'Y';
```

```sas
data VS1;
    set CDASH.VS;

 **** Variable level processing ;
❶  DOMAIN   = 'VS';
❷  USUBJID  = catx('.', STUDYID, SUBJID);
❸  VISITNUM = input(put(VISIT, $VISITNUM.), BEST.);

❹  [origin = Predecessor, do nothing];

❺  if       not missing(VISDAT)  then
        VSDTC = put(VISDAT, E8601DA.);
    else if not missing(VSDAT)   then
        VSDTC = put(VSDAT, E8601DA.);

❻  if VISIT = "VISIT 2 (WEEK 0)" then VSBLFL = 'Y';
run;
```

Mapping Specifications: **Value Level**

Source Sequence	Source Library	Source Dataset	Source Variable	Where Clause	Condition	Output	Map Sequence	Origin	Method	Comment	Code List	Target Library	Target Dataset	Target Variable	Target Data Type	Target Length	Significnat Digits
3	CDASH	VS		VS.VSTESTCD.EQ.DIABP	DIABP_VSPERF = 'Y'	Y	3	Assigned		DIABP		SDTM	VS	VSTESTCD	text	6	
3	CDASH	VS	DIABP_VSORRES	VS.VSTESTCD.EQ.DIABP	DIABP_VSPERF = 'Y'	Y	4	Predecessor				SDTM	VS	VSORRES	text	4	
3	CDASH	VS	DIABP_VSORRESU	VS.VSTESTCD.EQ.DIABP	DIABP_VSPERF = 'Y'	Y	5	Predecessor				SDTM	VS	VSORRESU	text	9	
3	CDASH	VS		VS.VSTESTCD.EQ.DIABP	DIABP_VSPERF = 'Y'	Y	6	Assigned		mmHg		SDTM	VS	VSSTRESU	text	9	
3	WORK		VSORRES	VS.VSTESTCD.EQ.DIABP	DIABP_VSPERF = 'Y'	Y	7	Convert			best.	SDTM	VS	VSSTRESN	float	8	0
3	WORK		VSSTRESN	VS.VSTESTCD.EQ.DIABP	DIABP_VSPERF = 'Y'	Y	8	Convert			4.0	SDTM	VS	VSSTRESC	text	6	
3	CDASH	VS	DIABP_VSPOS	VS.VSTESTCD.EQ.DIABP	DIABP_VSPERF = 'Y'	Y	9	Predecessor				SDTM	VS	VSPOS	text	7	
3	CDASH	VS		VS.VSTESTCD.EQ.HEIGHT	HEIGHT_VSPERF = 'Y'	Y	3	Assigned		HEIGHT		SDTM	VS	VSTESTCD	text	6	
3	CDASH	VS	HEIGHT_VSORRES	VS.VSTESTCD.EQ.HEIGHT	HEIGHT_VSPERF = 'Y'	Y	4	Predecessor				SDTM	VS	VSORRES	text	4	
3	CDASH	VS	HEIGHT_VSORRESU	VS.VSTESTCD.EQ.HEIGHT	HEIGHT_VSPERF = 'Y'	Y	5	Predecessor				SDTM	VS	VSORRESU	text	9	
3	CDASH	VS		VS.VSTESTCD.EQ.HEIGHT	HEIGHT_VSPERF = 'Y'	Y	6	Assigned		m		SDTM	VS	VSSTRESU	text	9	
3	CDASH	VS	HEIGHT_VSORRES	VS.VSTESTCD.EQ.HEIGHT	HEIGHT_VSPERF = 'Y'	Y	7	Derived	VS.VSSTRESN.item1			SDTM	VS	VSSTRESN	float	8	2
3	CDASH	VS	VSSTRESN	VS.VSTESTCD.EQ.HEIGHT	HEIGHT_VSPERF = 'Y'	Y	8	Convert			4.2	SDTM	VS	VSSTRESC	text	6	
3	CDASH	VS		VS.VSTESTCD.EQ.HEIGHT	HEIGHT_VSPERF = 'Y'	Y	9	Assigned		NULL		SDTM	VS	VSPOS	text	7	

❶
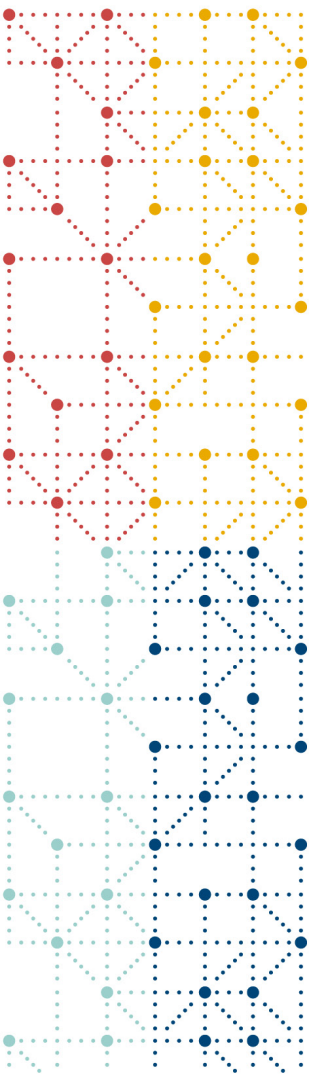
❷

```
data VS3;
    set CDASH.VS;

    if DIABP_VSPREF = 'Y' then do;
        VSTESTCD = 'DIABP';
        VSORRES  = DIABP_VSORRES;
        VSORRESU = DIABP_VSORRESU;
        VSSTRESN = 'mmHg';
        VSSTRESN = INPUT(VSORRES, BEST.);
        VSSTRESC = PUT(VSSTRESN, 4.0);
        VSPOS    = DIABP_VSPOS;
        OUTPUT;
    end;
```
❶

```
*** CONTINUE ***;

    if HEIGHT_VSPREF = 'Y' then do;
        VSTESTCD = 'HEIGHT';
        VSORRES  = HEIGHT_VSORRES;
        VSORRESU = HEIGHT_VSORRESU;
        VSSTRESN = 'm';
        VSSTRESN = INPUT(VSORRES, BEST.);
        VSSTRESC = PUT(VSSTRESN, 4.0);
        OUTPUT;
    end;
run;
```
❷

cdisc

Learnings so far

Machine-readable Metadata

Machine-readable Metadata

- CDISC 360 Enriched Metadata = Structural + Conceptual + Semantic + Process [Key to Automation]
  - Content is part of the standards (CDISC library)
  - ETL Metadata (mapping inference & derivation)

- System agnostic standards, concepts and elements
  - Can be consumed by any tool
  - Organization can build an automation engine their own way

- Iterations are needed to learn and evolve
  - Strong workstream collaboration: CDISC, Industry volunteers & Microsoft

WS1 & WS2

WS4

WS5 & WS6

Biomedical Concepts
Foundational Standards

cdisc
LIBRARY
formerly SHARE

cdisc
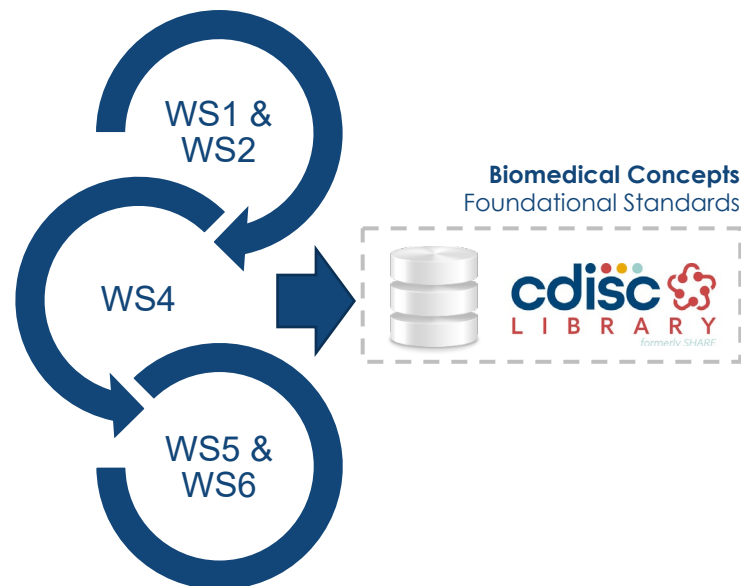
Thank You!

Bhavin Busa, *Vita Data Sciences*

Jianhui [Jimmy] Zhao, *Allergan*

Courtesy: Mahi Busa

cdisc