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
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
7. Learnings so far
Workstream 6 Introduction
CDISC 360 Workstreams

**Workstream 1 - ENHANCE STANDARDS**
Create concepts in knowledge graphs

**Workstream 2 - PUBLISH STANDARDS**
Load into library
Biomedical Concepts
Analysis Concepts
Foundational Standards

**Workstream 3**
Extend API’s

**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)

Transform concepts in machine readable form
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

Kaja Najumudeen, TalentMine, Jianhui Zhao, Allergan

TFL Automation Task Team Leads

Prasanna Murugesan, AstraZeneca, Stuart Malcolm, Frontier Science
Current State - without Concept-based Standards
Current State - without Concept-based Standards: CDASH to SDTM Execution

- **Manual Process**
- **Manual or Semi-automated Execution**

* SDTM Specifications
+ SAS Programs
  - SDTM Datasets
    - SDTM Define & aCRF
      - Submission
        - XPT files, Define, aCRF, SDRG

Specify | Build/Execute | Report

*Foundational Standards*
CDISC 360 Enriched Metadata
Machine-readable CDISC 360 Enriched Metadata

- Structural
- Conceptual
- Process
- Semantic
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

Reference: ‘CDISC 360 - The Journey so Far and the Road Ahead’, Peter Van Reusel, 28th April 2020
Linked Graph Model: Importing Concept-based Standards

Study Build of ODM.XML and Define.XML

ODM CRF Generated using Biomedical Concepts, Bindings, & Standards

ODM-based Vital Signs (VS) CRF

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

Study and Data Collection Designers

Design

- 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)
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

- **Source**: location (library name), datasets, processing sequence
- **Mapping**: fields needed to describe how source transits to target
- **Target**: location (library name), datasets, processing sequence, attributes (label, class, structure, purpose, etc.)

Dimension 2

- **Dataset Level**: Transit datasets from source to target
- **Variable Level**: Map variables from source to target
- **Value Level**: Map variables from source to target under different conditions
## Mapping Specifications: Dimension 1

<table>
<thead>
<tr>
<th>Source Sequence</th>
<th>Source Library</th>
<th>Source Dataset</th>
<th>Source Variable</th>
<th>Source Sequence</th>
<th>Mapping Sequence</th>
<th>Mapping Library</th>
<th>Mapping Dataset</th>
<th>Mapping Variable</th>
<th>Target Sequence</th>
<th>Target Library</th>
<th>Target Dataset</th>
<th>Target Variable</th>
<th>Target Description</th>
<th>Target Data Type</th>
<th>Target Length</th>
<th>Target Sorting Order</th>
</tr>
</thead>
</table>
## Mapping Specifications: Dimension 2

### Source
- **Library**
- **Dataset**

### Mapping
- **Source Sequence**
- **Source Library**
- **Source Dataset**
- **Source Variable**
- **Map Sequence**
- **Map Origin**
- **Map Method**
- **Comment**
- **Code List**
- **Target Sequence**
- **Target Library**
- **Target Dataset**
- **Target Variable**
- **Target Description**
- **Target Data Type**
- **Target Length**
- **Target Sorting Order**

### Target
- **Library**
- **Dataset**

### Dataset Level
- 1. CDASH
- 2. SDTM
- 3. SDTM
- 4. SDTM
- 5. SDTM

### Variable Level
- 4. Derived
- 4. Derived
- 4. Derived
- 4. Derived
- 4. Derived

### Value Level
- 4. Derived
- 4. Derived
- 4. Derived
- 4. Derived
- 4. Derived

---

**Example Entries:****

| Source Sequence | Source Library | Source Dataset | Source Sequence | Source Library | Source Dataset | Source Sequence | Source Library | Source Dataset | Source Sequence | Source Library | Source Dataset | Source Sequence | Source Library | Source Dataset | Source Sequence | Source Library | Source Dataset | Source Sequence | Source Library | Source Dataset | Source Sequence | Source Library | Source Dataset | Source Sequence | Source Library | Source Dataset |
|-----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1               | CDASH          | VS             | 3              | CDASH          | VS             | 4              | CDASH          | VS             | 4              | CDASH          | VS             | 4              | CDASH          | VS             | 4              | CDASH          | VS             | 4              | CDASH          | VS             | 4              | CDASH          | VS             | 4              | CDASH          | VS             | 4              | CDASH          | VS             |
| 2               | SDTM           | DM             | 4              | SDTM           | DM             | 4              | SDTM           | DM             | 4              | SDTM           | DM             | 4              | SDTM           | DM             | 4              | SDTM           | DM             | 4              | SDTM           | DM             | 4              | SDTM           | DM             | 4              | SDTM           | DM             | 4              | SDTM           | DM             |
| 3               | SDTM           | SV             | 4              | SDTM           | SV             | 4              | SDTM           | SV             | 4              | SDTM           | SV             | 4              | SDTM           | SV             | 4              | SDTM           | SV             | 4              | SDTM           | SV             | 4              | SDTM           | SV             | 4              | SDTM           | SV             | 4              | SDTM           | SV             |
| 4               |                |                |                |                |                |                |                |                |                |                |                |                |                |                |                |                |                |                |                |                |                |                |                |                |                |                |                |                |                |

---

**Legend:**
- **Subset Condition:**
- **Pre Processing:**
- **Join Type:**
- **Join Timing:**
- **Merge Key:**

---

**Notes:**
- Dataset Level: CDASH, SDTM, VS
- Variable Level: Derived, Assigned
- Value Level: Derived, Assigned

**Source Variable Examples:**
- CDASH: VS, SDTM
- SDTM: DM, SV
- Derived: VS, SDTM
- Assigned: VS, SDTM

---

**Target Variable Examples:**
- Vital Signs Test Short Name: text, 6
- Nutritional Data: text, 8
- Nutritional Data: float, 8

---

**Target Description Examples:**
- Nutritional Data: text, 8
- Nutritional Data: float, 8

---

**Library Examples:**
- CDASH: VS
- SDTM: DM, SV
- Derived: VS, SDTM
- Assigned: VS, SDTM

---

**Dataset Examples:**
- SDTM: DM, SV
- Derived: VS, SDTM
- Assigned: VS, SDTM

---

**Sorting Order Examples:**
- Nutritional Data: float, 8
- Nutritional Data: float, 8
Mapping Specifications: Dataset Level

---

<table>
<thead>
<tr>
<th>Source Sequence</th>
<th>Source Library</th>
<th>Source Dataset</th>
<th>Subset Condition</th>
<th>Pre Processing</th>
<th>Join Type</th>
<th>Join Timing</th>
<th>Merge Key</th>
<th>Target Sequence</th>
<th>Target Library</th>
<th>Target Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CDASH</td>
<td>VS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5</td>
<td>SDTM</td>
<td>VS</td>
</tr>
<tr>
<td>2</td>
<td>SDTM</td>
<td>DM</td>
<td></td>
<td>TARGET</td>
<td>PRE</td>
<td>USUBJID</td>
<td></td>
<td>5</td>
<td>SDTM</td>
<td>VS</td>
</tr>
<tr>
<td>3</td>
<td>SDTM</td>
<td>SV</td>
<td></td>
<td>TARGET</td>
<td>PRE</td>
<td>USUBJID, VISITNUM</td>
<td></td>
<td>5</td>
<td>SDTM</td>
<td>VS</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td>SORT</td>
<td></td>
<td>USUBJID, VISITNUM, VSDTC</td>
<td></td>
<td>5</td>
<td>SDTM</td>
<td>VS</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td>SORT</td>
<td></td>
<td>USUBJID, VSTESTCDE, VISITNUM, VSDTC</td>
<td></td>
<td>5</td>
<td>SDTM</td>
<td>VS</td>
</tr>
</tbody>
</table>

---

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

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

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

4. `... Sequence 3, 4`

5. `proc sort data=VS4;`  
   `by` USUBJID VSTESTCDE VISITNUM VSDTC;
   `run;`

6. `data` SDTM.VS;
   `set` VS4;
   `by` USUBJID VSTESTCDE VISITNUM VSDTC;
   `/******** variable level: Source Sequence = 6  
     *********/`
   `run;`
Mapping Specifications: Variable Level

```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

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;

*** CONTINUE ***;

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

Bhavin Busa, *Vita Data Sciences*
Jianhui [Jimmy] Zhao, *Allergan*