Mapping a Veeva SDS specification to ODM-XML

Jørgen Mangor Iversen, Novo Nordisk A/S, Copenhagen, Denmark

Abstract

As Veeva EDC grows increasingly popular across the industry, more and more sponsors become acquainted with its Study Design Specification (SDS) document. The focus here is on what information is inside the SDS and how it is related to the CDISC ODM-XML standard and serves as inspiration on how to map an SDS to a generic ODM-XML. Topics covered include the level of transformations, a discussion of internal and external keys, data types, vendor specific data, edit checks, and ODM-XML data not found in the SDS. The scope is solely the definitions of library CRF forms and does not include any other data that may be stored in ODM-XML format.

Introduction

One of the most valuable ideas, not only in the pharma industry, but in any industry driven by IT systems, is the ability to have both data and metadata flow freely between systems via software integrations. Such integrations generally consist of two parts: The transfer of data and conversion between data formats. In this context, this paper deals with conversion of data formats between two systems of growing popularity: Veeva EDC and Study Builder[1]. Both systems contain definitions of CRF Forms[2] and controlled terminology (CT). But rather than maintaining two sets of the same definitions and thus creating a synchronization problem, the better strategy is to create one from the other via a system integration.

Strategy

The overall strategy here is to use Study Builder as intended as a metadata repository, maintaining a full set of standard Forms, a full set of CT for global dictionaries[3] as well as sponsor defined CT, and to use Veeva EDC to create Forms for data collection in clinical studies and trials[4], as well as collecting the actual clinical data and passing it on for further processing.
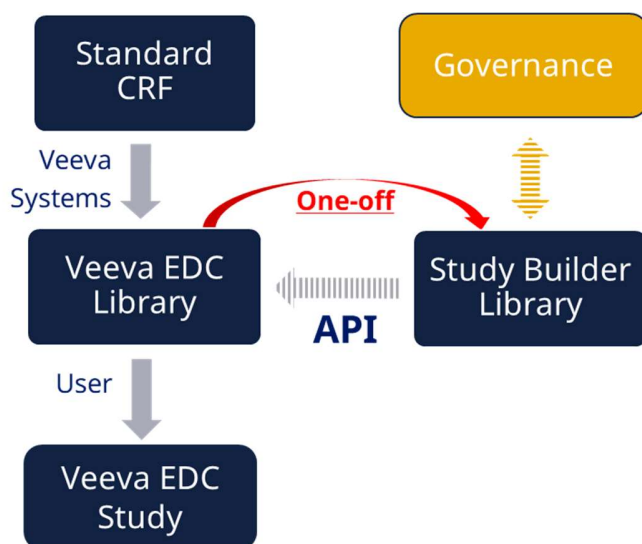
A subset of the standard Forms has been created in Veeva EDC. Those needs to be imported into Study Builder to form the basis of the Library of Forms and to be the subject of governance from within Study Builder. A future API will allow transfer of Library forms from the Study Builder Library to the Veeva EDC Library. Study Forms will then be created inside Veeva EDC from the Veeva EDC Library Forms.

The only way of exporting CRF metadata from Veeva EDC is a Veeva Study Design Specification (SDS), and the only way to import CRF metadata into Study Builder is an ODM-XML document. This is the rationale for converting an SDS document to an ODM-XML document. ODM-XML is a well-established standard from CDISC, however somewhat under-implemented by software vendors.

---

[1] Eventually also Open Study Builder
[2] CRF and Forms are used interchangeably for any definition of a page to collect clinical data
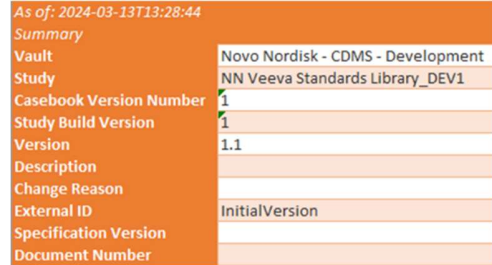[3] CDISC, SNOMED, MedDRA, MED-RT, UNII, LOINC, UCUM
[4] Study and trial are used interchangeably for any collection of clinical data in the pharma industry

What is a Veeva EDC Document

The Veeva SDS is basically an Excel document divided into many individual sheets (21) documenting the information necessary to define a setup for a study. In Veeva EDC the Library is simply a special instance of a study. For defining the Forms, only a small subset of all these sheets is needed:

- Summary
- Schedule – Tree
- Form Definitions
- Code lists
- Unit code lists
- Rules contain edit checks only indirectly in scope

What is an ODM-XML Document

An ODM-XML document is a file format defined by CDISC, formally defined as a superset of both CRF definitions and data definitions (define.xml). An ODM-XML contains in its pure form all metadata for a collection of CRFs. Beyond the CRF's themselves, it can contain much more, such as:

- Visit structure
- User information
- Reference data
- Clinical data
- Digital signature

```xml
<?xml version="1.0" encoding="UTF-8"?>
<ODM FileOID          = "T_egALL_FORMS"
    FileType         = "Snapshot"
    CreationDateTime = "2024-04-18T16:14:06"
    AsOfDateTime     = "2024-04-09T11:28:11"
    ODMVersion       = "1.3.2"
    Originator       = "Novo Nordisk"
    SourceSystem     = "SAS Enterprise Guide: sds2odm.sas"
    xmlns:xml        = "http://www.w3.org/XML/1998/namespace"
    xmlns            = "http://www.cdisc.org/ns/odm/v1.3">
<Study OID="S_evALL_FORMS">
  <GlobalVariables>
    <StudyName>NN_Veeva_Standards_Library_DEV1</StudyName>
    <StudyDescription>NN Veeva Standards Library_DEV1</StudyDescription>
    <ProtocolName>NN_Veeva_Standards_Library_DEV1</ProtocolName>
  </GlobalVariables>
  <BasicDefinitions>
    <MeasurementUnit OID="Unit 1" Name="DOSE STEP">
      <Symbol>
        <TranslatedText>Unit 1</TranslatedText>
      </Symbol>
    </MeasurementUnit>
    <MeasurementUnit OID="mmol/L" Name="MMOL_L">
      <Symbol>
        <TranslatedText>mmol/L</TranslatedText>
      </Symbol>
    </MeasurementUnit>
  </BasicDefinitions>
  <MetaDataVersion OID="NN_Veeva_Standards_Library_DEV1" Name="NN Veeva
Standards Library_DEV1">
```

CRF Conversions

The conversion of CRF related information from the Veeva SDS document to an ODM-XML document is broken down into segments after the issues found.

Global Data

The Summary sheet contains good candidates for high-level ODM identifiers such as StudyName and

MetaDataVersion OID. The Schedule – Tree sheet contains possible candidates for other high-level ODM identifiers such as FileOID and StudyOID. The actual conversion may use these candidates for high-level ODM identifiers, or may extract other values i.e. from file names etc.

Form Structure

The whole CRF structure is found in a single sheet Form Definitions, wherein Forms, Sections, and questions are interleaved into the same sheet, so relationships between Forms, Sections, and Questions must be deduced across rows. These map straightforward into ODM elements Forms, ItemGroups, and Items. In ODM-XML documents Itemgroups have an attribute indication whether the ItemGroup is mandatory for a given Form (this may vary across Froms). This information is not explicitly present in the SDS Form Definitions sheet. I have chosen to infer this information (with care) from the column Display Format, where a value of 'Form' is interpreted as the section is displayed on the Form.

Order Numbers

The sequence of elements within an ODM-XML document is a very important feature to describe a Form accurately. In ODM-XML this is done via OrderNumber attributes wherever ordering is necessary. Such an explicit ordering information is not present in the Veeva SDS document. There the ordering information is given implicitly by the sequence of rows in sheets containing data to be ordered. For any converter, this means that great care must be taken when sorting data, i.e. when looking up references across sheets. As previously mentioned, Forms, Sections, and Questions are interleaved in the same sheet without any other ordering indication beyond row sequence.

ODM-XML has no requirements to OrderNumber beyond ordering, but humans like to start over for each form and section. For this rather simple reason, the converter re-starts the ordering from 1 with every change of identifier at a higher level. So ItemGroups are numbered from 1 within each Form, an Items are numbered from 1 within each ItemGroup. The SDS document displays row colors on all sheets to enhance human readability, but row band colors are not machine readable by the Excel importer used t build the converter tool.

Form Name	Item Group Name	Item Name
MH		
MH	MH	
MH	MH	MHCAT
MH	MH	MHTERM_ALLERGY
MH	MH	MHTERM_ALLEROTH
MH	MH	MHTERM_BREAST
MH	MH	MHTERM_BREASOTH
MH	MH	MHTREM_CVDP
MH	MH	MHTERM_CVDPOTH
MH	MH	MHTERM_CAS
MH	MH	MHTERM_CASOTH
MH	MH	MHTERM_STROKE
MH	MH	MHTERM_STROKOTH
MH	MH	MHTERM_PERI
MH	MH	MHTERM_PERIOTH
MH	MH	MHTERM_AF
MH	MH	MHTERM_AFOTH
MH	MH	MHTERM_CVD
MH	MH	MHTERM_CVDOTH
MH	MH	MHTERM_DIAB
MH	MH	MHTERM_DIABOTH
MH	MH	MHTERM_DYSLI

Identifiers

The Veeva SDS document uses the Name column as the primary key for all elements within a sheet. The Form Definitions also has a column External ID, which is a strong candidate for ODM OIDs, depending on the choices made when creating the Forms in Veeva EDC to be converted into OMD-XML. The main advantages of External ID are that it is independent of internal SDS references, and that it imposes fewer restrictions on the value content. The External ID may contain spaces and periods, where the Name may not.

It quickly becomes apparent that the main issue in any converter including this one is the alignment of keys between the systems being integrated. This includes matching existing content in the receiving system when importing, enabling other software to compare CRFs, CT, and other data between the systems involved, and also for mappings between SDS and ODM-XML. Traceability via mapped or identical keys is crucial for validating correctness of the conversion.

Study Builder in particular, is very particular regarding the value of keys for matching previous content. This may seem like a general statement valid for all systems, but for Study Builder it also applies when importing new content.

- MeasurementUnit OID must match internal values exactly, always! New units cannot be created when importing an ODM-XML
- CodeList OID must match internal values exactly when referring existing code lists
- CodeListItem must match an OID and a Name in the osb name space exactly when referring existing terms

Consequently, any ODM-XML for import must have prior knowledge of Study Builder's preexisting contents. When Name and External ID are set by Veeva SDS, a mapping prior to data conversion must occur to translate them into Study Builder OIDs. This may not apply when importing the same converted ODM-XML into other systems.

Data Types

Conversion of most data types is straightforward. A set of simple rules is set up.
- All Items having values in the Decimal column are converted into a floating-point type
- All Items having a unit are converted into an integer type, except if the floating-point conversion rule is already in effect
- Code lists suffer from an insufficient definition of data types. In the Form Definitions sheet, they have the data type of Codelist, and in the Codelists sheet, the data type is not defined. This has been resolved through a simple algorithm, assuming that the values of code list terms consist of the only possible allowed in the code list. The code list term values are thus inspected
  - If any value of terms in a code list contains any alphabetic character, they are data type text
  - If any remaining value contains a decimal point, they are data type floating-point
  - In all other cases they are data type integer, assuming that dates and date times are not the subject of CT
- SDS has a label data type, supported by Study Builder as a comment data type, which is treated as an extension of the ODM-XML CDISC specification
- Apart from these rules only minor changes to case and names apply

SDS Data type	ODM Data type
Has decimals	float
Is a unit	integer (float if dec.)
Is a code list	Inspect terms for alphabetic characters
Boolean	boolean
Date	date
Date/Time	datetime
Number	integer
Text	text or string
Form Link	text or string
Label	comment

Mappings

The actual mapping between the 2 documents formats is either straight forward 1:1 datapoints only with minor changes such as case change or the odd prefix or is quite complex with a lot of processing and database lookups. The latter is mostly due to peculiarities of importing an ODM-XML into Study Builder, requiring the fore mentioning keys to match existing contents to avoid creating a lot of duplicate and overlapping controlled terminology.

The Tables describing the mappings themselves are not exhaustive, leaving details such as algorithms and ODM-XML syntax to implementations for specific systems.
- Adding prefixes to OIDs distinguishing between Forms, ItemGroups, Items, Units, and Codelists has been performed but is not reflected in the tables
- The <Study> tag attribute are populated from several SDS sheets, indicated by the syntax of Excel cross sheet references
- The Study Builder name space attributes can be omitted for other systems
- Additional text for guidance and instructions have not been entered consistently for Forms, ItemGroups, and Items in the SDS document provided for this software integration

SDS: Form Definitions	ODM: ItemGroupDef		SDS: Form Definitions	ODM: ItemDef
Item Group Name	@OID		Item Name	@OID
Item Group Label[1]	@Name		Item Name	@Name
IG Rep	@Repeating		Data Type	@DataType[2]
Description[1]	<Description>		Length	@Length[2]

SDS: Form Definitions	ODM: FormDef			Decimal	@SignificantDigits[2]
Form Name	@OID			Label[1]	<Question>
Form Label[1]	@Name			Codelist	<CodeListRef>[2]
Repeats	@Repeating			Unit Codelist	<MeasurementUnitRef>[2]
Form Short Label[1]	@osb:instruction			Hint Label[1]	<Description>
Hover Help[1]	@osb:sponsorinstruction			Hover Help[1]	@osb:instruction
Description[1]	<Description>			Description[1]	@osb:sponsorinstruction

SDS: Form Definitions	ODM: ItemGroupRef		SDS: Form Definitions	ODM: ItemRef
Item Group Name	@ItemGroupOID		Item Name	@ItemOID
Display Format = 'Form'	@Mandatory[2]		Required	@Mandatory
[calculated]	@OrderNumber[2]		[calculated]	@OrderNumber[2]

SDS: Schedule - Tree	ODM: Study		SDS: Codelists	ODM: CodeList
Event Group Name	@FileOID		[calculated] or Name	@OID[2]
datetime()	@CreationDateTime		[calculated]	@Name[2]
=Summary'!A1	@AsOfDateTime		[calculated from ItemDef]	@DataType[2]
Event Name	@Study_OID		Name	@SASFormatName
=Summary'!B4	<GlobalVariables>		Description[1]	<Description>

SDS: Unit Codelists	ODM: MeasurementUnit		SDS: Codelists	ODM: CodeListItem
[calculated]	@OID[2]		Choice Code[1]	@CodedValue
Name	@Name		[calculated][2]	@OrderNumber
Choice Label[1]	<Symbol>		Choice Label[1]	<Decode>
"1.0"	@Version			

1) Requires HTML encoding due to contents not allowed in XML
2) Requires additional processing

Vendor Specific Data

As only a few of all the sheets in the SDS document is used, most of the SDS content is considered vendor specific metadata beyond the CRF metadata. However, the XML definition allows for vendor specific name spaces to be added, a technique utilized by Study Builder already. Furthermore, the ODM specification allows for Alias tags to be added in many places, which can be utilized for creating vendor specific Content attributes. Both are perfectly valid options to carry vendor specific data, as both are safe to omit without violating the ODM-XML validity.

Name Space

<any_tag veeva:example="data value">

Alias

<Alias Context="Veeva"> Name="data value"/>

Should further vendor specific data be chosen to be included in the conversion of the Veeva SDS document into ODM-XML, a decision has to be made regarding adding a vendor name space or adding Alias tags to the resulting ODM-XML document.

Edit Checks

```
#define AMERICAN_INDIAN @Form.DM.AMERICAN_INDIAN
#define ASIAN @Form.DM.ASIAN
#define BLACK @Form.DM.BLACK
#define NATIE_HAWAIIAN @Form.DM.NATIE_HAWAIIAN
#define WHITE @Form.DM.WHITE
#define NOT_REPORTED @Form.DM.NOT_REPORTED

AMERICAN_INDIAN != true
&&
ASIAN != true
&&
BLACK != true
&&
NATIE_HAWAIIAN != true
&&
WHITE != true
&&
WHITE != true
&&
NOT_REPORTED != true
```

One example of vendor specific data to be included is the edit checks found in the Rules sheet. These checks are formulated in a Veeva specific language, Veeva Clinical Query Language (CQL), which seems to be machine generated. A syntax for this language must this exist or can fairly easily be inferred. This leads to the idea that a cross compiler can be written to convert CQL to and from any rule definition language.

Data errors

During development of the converter, a number of data errors were encountered.

- Some of the identifiers intended to be ODM OIDs contained spaces, particularly Veeva Names
- Some of the explanatory and longer text values in questions on Forms contains double quates, which is a direct violation of the specification of XML, which has Attribute values in double quotes. This was resolved through HTML encoding of all text values
- Some text values had text quoting 'Unknown' instead of ASCII quoting 'Unknown'. This is a very subtle issue distinguishing between quotation marks that are slanted (and in fact accents, not quotation marks) and straight vertical proper single quotation marks. This was resolved through HTML encoding of all text values
- Some text values had binary characters in them, notably hexadecimal values in the range 01X to 1FX, not allowed in the XML specification.

Examples
OID="NN Veeva Standards Library_DEV1"
Report time as "?" if diary response is 'Unknown' for time of episode.
Peripheral artery stenosis (≥50% stenosis)

These issues may not be characteristics of Veeva EDC but may very well stem from the originating system supplying the standard Forms created in Veeva EDC. The lesson here is to observe the dangers of using copy/paste of data across systems having different requirements.

ODM Omissions

In summary, issues have been found where the SDS document cannot directly and simply supply all the data for an ODM-XML document. Work-arounds have been implemented for some of them, but other have simply been omitted as they are optional, even if they serve a purpose.

- Explicit indication of whether a section is mandatory. The rule to infer this may be confirmed or denied by Veeva Systems
- Explicit ordering of sections and question has been inferred by the ordering of rows
- Explicit data types for code lists are not part of the SDS document, but can be inferred by examining the values of the code list terms themselves
- Explicit data types for questions having units are not part of the SDS document, but can be inferred by the presence of decimals
- Multiple languages for questions and descriptions throughout are not supported by the SDS document, why English has been assumed for all

Conclusion

The main conclusion is that mapping a Veeva SDS document to a CDISC ODM-XML document is doable, and not really that hard. A mapping allows for conversion between the two, both ways, and the number and type of transformations from one to the other are not substantial. While discrepancies between the two document formats are bridgeable, any conversion software must be aware of the smaller details regarding relationships, ordering, and data types. Extension into vendor specific data can be implemented as either XML name space or ODM extensions such as Alias, a decision just has to made on the mechanism of choice. As ODM-XML has recently been released in a new version 2.0, this offers a golden opportunity for both Veeva EDC and Study Builder as well as other relevant software vendors to align the support for the ODM-XML standard to align for better software integrations to the benefit for common customers.

Contact Information

Your comments, only minor changes to case and names of and questions are valued and encouraged. Contact the author at:

Author Name: Jørgen Mangor Iversen
Company: Novo Nordisk A/S
Email: jqiv@novonordisk.com
Website: https://www.novonordisk.com/

Brand and product names are trademarks of their respective companies.