CDISC Library-XML Specification Version 1.0.0

Prepared by

Sam Hume

Notes to Readers

- This is the CDISC Library-XML ODM and Define-XML extensions specification.
- The Library-XML extension specification is an informational document.

Revision History

Date	Version	Summary of Changes
2016-02-15	DRAFT 1.0	Initial draft for review.
2018-03-01	FINAL 1.0 SHARE	Final version
2019-04-26	Draft 1.0	Library version draft. Mapping instructions in Appendix B.
2021-06-17	Draft 1.0	Based on implementation

Table of Contents

1 INTRODUCTION....4

1.1 Purpose of this document....4
1.2 Library-XML Release Package....4
1.3 CDISC....4
1.4 Relationships to Other CDISC Standards....4
1.4.1 Operational Data Model (ODM)....4
1.4.2 Define-XML....5
1.4.3 Study Data Tabulation Model (SDTM) Implementations....5
1.4.4 Standard for Exchange of Nonclinical Data (SEND)....5
1.4.5 Analysis Data Model (ADaM) Implementations....6
1.4.6 Clinical Data Acquisition Standards Harmonization (CDASH)....6

2 ABBREVIATIONS AND REFERENCES....7

2.1 Abbreviations and Terms....7
2.2 References....8

3 CONFORMITY AND GENERAL ISSUES....9

3.1 File Conformity....9
3.2 Extensions....9
3.3 Library-XML Document Structure....10
3.4 Validation of a Library-XML document....13

4 USING LIBRARY XML....14

4.1 Library-XML Extensions for CDASH....14
4.1.1 Example of the Library-XML Extension used for CDASH....16
4.2 Library-XML Extensions for ADaM, SDTM and SEND....17
4.2.1 Example of the Library-XML Extension used for SDTM....19
4.3 Library-XML Extensions for API Support....19
4.3.1 Example of the ExternalCodeList Use with API Support....19
4.4 Library-XML use of Alias....19
4.4.1 Example of the Alias as a child element of CodeList....20
4.4.2 Example of the Alias as a child element of ItemDef....20

5 SPECIFICATION....20

5.1 Library-XML Scope....20
5.2 Library-XML Structure....20
5.3 Library-XML Specification Details....23
5.3.1 XML Header....23
Example XML Header....23
5.3.2 ODM Element....23
5.3.3 MetaDataVersion Element....25
5.3.3.1 Example MetaDataVersion Element....25
5.3.3.2 Alias Element....26
5.3.4 ItemGroupDef Element....26
5.3.4.1 mdr:*Class*....27
5.3.5 ItemDef Element....28
5.3.5.1 mdr:AltCodeListRef....29
5.3.5.2 mdr:Prompt....29
5.3.5.3 mdr:CRFCompletionInstructions....30

5.3.5.4 mdr:ImplementationNotes.....30
5.3.5.5 mdr:MappingInstructions.....30
5.3.5.6 mdr:Definition.....31
5.3.5.7 mdr:CDISCNotes.....31
5.3.6 CodeList Element.....31
5.3.6.1 ExternalCodeList.....32
6 ACKNOWLEDGMENTS.....33
7 APPENDICES.....33
7.1 APPENDIX A: XML SCHEMA.....33
7.2 APPENDIX B: LIBRARY-XML PROGRAMMING SPECIFICATIONS.....35
7.3 APPENDIX C: REPRESENTATIONS AND WARRANTIES, LIMITATIONS OF LIABILITY, AND DISCLAIMERS 38

1 Introduction

1.1 Purpose of this document

This specification describes the CDISC Library-XML ODM and Define-XML extensions. Library-XML captures the additional metadata needed to represent standards repository metadata for CDISC standards such as ADaM, CDASH, SDTM and SEND. The purpose of Library-XML is to enable Library and other metadata repositories (MDR) to publish a broader set of metadata that more completely describes the standards. It supports the exchange of metadata between an MDR like CDISC Library and software applications that seek to use the standards metadata. The extension provides a common way to specify this content within ODM and Define-XML, as well as providing a mechanism for schema validating this content. The Library-XML ODM and Define-XML extensions are not formal standards, but instead represent the format for publishing standards metadata from the CDISC Library MDR.

The examples provided in this specification were created based on the SHARE 1.0 implementation. The CDISC Library 1.0 implementation provides a version of this media-type adapted for the CDISC Library implementation. In this draft the programming specifications provided in Appendix B take precedence over the content in the examples.

1.2 Library-XML Release Package

The Library-XML release package includes:

- Library-XML v1.0 specification
- Library-XML v1.0 schemas
- Library-XML v1.0 examples

1.3 CDISC

The Clinical Data Interchange Standards Consortium (CDISC) is a non-profit organization whose mission is to develop and support global, platform-independent data standards that enable information system interoperability to improve medical research and related areas of healthcare.

CDISC Library, a cornerstone of the CDISC technical roadmap, is a global electronic repository for developing, integrating and accessing CDISC metadata standards in electronic format. CDISC Library is envisioned to help users find, understand and use rich metadata and controlled terminologies relevant to clinical studies more efficiently and consistently, and to improve integration and traceability of clinical data from protocol through analysis.

The FDA has collaborated with CDISC since its founding in order to standardize the content and structure of clinical trials and non-clinical study data for regulatory submission. CDISC sponsors and members represent more than 430 companies active in the research and development of regulated health-related products.

1.4 Relationships to Other CDISC Standards

1.4.1 Operational Data Model (ODM)

Library-XML provides an extension based on the CDISC Operational Data Model (ODM) XML schema. ODM is a vendor-neutral, platform-independent format for the interchange and

archival of clinical study data. The model includes the clinical data along with its associated metadata, administrative data, reference data and audit information. ODM includes all of the information that needs to be shared among different software systems during study setup, operation, analysis, and submission or for long term retention as part of a study archive. ODM has been embraced by a broad range of clinical development organizations, and a number of vendors provide software applications and tools that use ODM. The current version of the ODM standard is available at http://www.cdisc.org/odm.

One of the features of the ODM is a standardized mechanism for defining schema extensions to provide functionality needed to support interchange requirements for specialized use cases. These extensions follow the guidelines for Vendor Extensions provided in the ODM specification and comply with the W3C XML Schema 1.0 specification.

While this document is intended to be understandable to readers with minimal technical knowledge of the ODM and XML, knowledge of this document alone is not a substitute for knowledge of the ODM. This document should be used in close concert with the current version of the ODM specification as well as current versions of the relevant CDISC data and metadata standards. The ODM specification package, including the relevant schemas, is available online at http://www.cdisc.org/odm.

1.4.2 Define-XML

The Library-XML also provides a Define-XML extension that is intended for use with the Define-XML standard. The Define-XML v2.1.0 specification describes a model that defines CDISC SDTM, SEND and ADaM datasets as well as accommodating any other tabular dataset structure. One of the key benefits to FDA reviewers is that Define-XML provides both a machine readable format for use by the various FDA software applications and, through the provision of an XSL stylesheet, a browser-based report describing the contents of clinical study datasets.

Define-XML v2.1.0 can be used to transmit metadata for any tabular dataset, including the following CDISC standards: SDTM, ADaM, SEND, and SDTM Implementation Guide for Medical Devices. Define-XML v2.1 and later are recommended for use with Library-XML.

1.4.3 Study Data Tabulation Model (SDTM) Implementations

Library-XML extends the metadata represented in Define-XML to transmit additional SDTM metadata based on the SDTM Implementation Guide Versions 3.1.2 and higher, and other IGs based on SDTM. SDTM defines a standard structure for case report form data tabulations that are required to be submitted as part of a product application to the FDA. The CDISC SDTM is used to submit clinical trial and non-clinical study data for product applications across all therapeutic areas. The current version of the SDTM and SDTMIG standards are available at http://www.cdisc.org/sdtm.

1.4.4 Standard for Exchange of Nonclinical Data (SEND)

Library-XML extends the metadata represented in Define-XML to transmit additional SEND metadata based on the SEND Implementation Guide versions 3.0 and higher. SEND is an implementation of SDTM for non-clinical studies, which specifies a way to present non-clinical data in a consistent format. This type of study is typically related to animal testing as part of pre-clinical (pre-Phase 1) trials. SEND datasets may be included as part of a product application to the FDA.

The current version of the SEND Implementation Guide is available at http://www.cdisc.org/send.

1.4.5 Analysis Data Model (ADaM) Implementations

Library-XML extends the metadata represented in Define-XML to transmit additional ADaM metadata based on the ADaM Implementation Guide Versions 1.0 and higher, and other IGs based on ADaM. ADaM defines a standard for analysis datasets that are to be submitted in addition to the data tabulation (CDISC SDTM) datasets as described above in Section 1.4.3. The current version of the ADaM and ADaMIG standards are available at http://www.cdisc.org/adam.

1.4.6 Clinical Data Acquisition Standards Harmonozation (CDASH)

Library-XML extends the metadata represented in ODM to transmit additional CDASH metadata based on the CDASH version 1.1 and higher. CDASH version 1.1 was developed with participation from organizations in all three ICH regions (US, Europe and Japan). The standard describes the basic recommended data collection fields for 18 domains, including demographics, adverse events, and other common domains that are common to most therapeutic areas and phases of clinical research. In addition, CDASH V 1.1 includes implementation guidelines and best practice recommendations, regulatory references and other information on the CDASH project.

The current version of the CDASH standard and User Guide is available at http://cdisc.org/cdash.

2 Abbreviations and References

2.1 Abbreviations and Terms

ADaM	Analysis Data Model - developed by CDISC.
ADaMIG	Analysis Data Model Implementation Guide – developed by CDISC
CRF	Case Report Form
CRO	Contract Research Organization
CRT	Case Report Tabulation
define.xml	An instance of a Define-XML document. See Define-XML.
Define-XML	Define-XML is the CDISC standard for transmission of metadata for SDTM, SEND, ADaM, and any other tabular datasets.
eCTD	Electronic Common Technical Document
FDA	United States Food and Drug Administration
ICH	International Conference on Harmonization of technical requirements for registration of pharmaceuticals for human use.
ODM	Operational Data Model – developed by CDISC as an XML format for the transmission and archival of clinical trials data and metadata.
OID	ODM element identifier.
PDF	Portable Document Format – an open standard for document exchange developed by Adobe Systems
Dataset-XML	Dataset-XML supports the interchange of tabular clinical research data using ODM-based XML technology.
SDTM	Study Data Tabulation Model - developed by CDISC for the purpose of submitting Study Data Tabulations to the United States Food and Drug Administration.
SDTMIG	Study Data Tabulation Model Implementation Guide – developed by CDISC
SDTMIGMD	Study Data Tabulation Model Implementation Guide for Medical Devices – developed by CDISC
SEND	Standard for Exchange of Non-clinical Data – developed by CDISC
CDISC Library	The CDISC standards metadata repository
URI	Uniform Resource Identifier - a string of characters used to identify a resource on the internet

URL	Uniform Resource Locator
W3C	World Wide Web Consortium
XLink	XML Linking Language – developed by the W3C
XML	Extensible Markup Language - developed by the W3C
XPT	SAS Transport (XPORT) Format – an open standard for data transmission developed and maintained by SAS (see http://support.sas.com/techsup/technote/ts140.html)
XSD	XML Schema Definition (see http://en.wikipedia.org/wiki/XML_Schema_(W3C))
XSL	Extensible Stylesheet Language – developed by the W3C for the purpose of transforming and formatting XML documents

2.2 References

The documents referenced during the development of this Library-XML Reference may be accessed via the links provided below.

- CDISC website
  http://www.cdisc.org
- ODM Version 1.3.2
  http://www.cdisc.org/odm
- Define-XML Version 2.1
  http://www.cdisc.org/define-xml
- SDTM - Study Data Tabulation Model (SDTM) Final Version 1.2 and higer
  http://www.cdisc.org/sdtm
- SDTM-IG - CDISC SDTM Implementation Guide Version 3.1.2 and higher
  http://www.cdisc.org/sdtm
- SDTM-MSG – SDTM Metadata Submission Guidelines V1
  http://www.cdisc.org/sdtm
- ADaM - CDISC Analysis Data Model (ADaM) Version 2.1
  http://www.cdisc.org/adam
- ADaM-IG - CDISC Analysis Data Model (ADaM) Implementation Guide Version 1.0
  http://www.cdisc.org/adam
- SEND-IG - CDISC SEND Implementation Guide Version 3.0 and higher
  http://www.cdisc.org/send
- Controlled Terminology
  http://www.cancer.gov/cancertopics/cancerlibrary/terminologyresources/cdisc
- XML Schema Validation for Define.xml White Paper
  http://www.cdisc.org/define-xml
- FDA eCTD Guidance - Electronic Common Technical Document
  http://www.fda.gov/downloads/Drugs/GuidanceComplianceRegulatoryInformation/Guidances/UCM072349.pdf
- FDA Study Data Specifications
  http://www.fda.gov/downloads/ForIndustry/DataStandards/StudyDataStandards/UCM312964.pdf
- FDA Study Data Standards Page

http://www.fda.gov/forindustry/datastandards/studydatastandards/default.htm

3 Conformity and General Issues

This section supplements the corresponding section, "General Issues", of the ODM v1.3.2 specification.

All conformity requirements described in the ODM v1.3.2 specification are also applicable to Library-XML files unless stated otherwise.

3.1 File Conformity

The namespace URI for version 1.0.0 of Library-XML is:
http://www.cdisc.org/ns/library-xml/v1.0.

Throughout this document, the following conventions are used for namespaces:

- ODM elements and attributes are in the default namespace (i.e. they have no namespace prefix)
- Define-XML elements use the namespace prefix "def"
- Define-XML attributes use the namespace prefix "def" only if they appear within ODM elements
- Library-XML elements use the namespace prefix "mdr"
- Library-XML attributes use the namespace prefix "mdr" only if they appear within ODM elements

Note that these namespace prefixes are used throughout this document and are recommended as best practice both to make it easier for users to understand and implement Library-XML, and to aid in the comparison of documents.

Unlike other extensions like Define-XML, any ODM or Define-XML elements and attributes not described in this document may be included in Library-XML. The XML must be valid within the ODM, Define-XML, or Library-XML namespaces. Library-XML intends to support interchange for a broad range of MDR metadata, and is thus less restrictive than Define-XML use in the context of a regulatory submission.

Deprecated elements or attributes are not valid for use and are considered errors.

3.2 Extensions

The Library-XML schema permits vendor extensions, as defined in the ODM v1.3.2 specification, to the elements defined in this specification. These extensions may take the form of CDISC-created extensions, such as Define-XML, or vendor extensions. Any XML not explicitly specified as part of Library-XML v1.0.0, ODM v1.3.2, or Define-XML v2.1 is considered an extension. Extensions have no implied meaning with respect to the Library-XML specification; the sender and receiver must agree on a meaning between themselves. That is, *Library–XML* files that use extensions are not wrong, but instead the extensions may be ignored unless the sender and receiver have agreed otherwise.

Requirements for vendor extensions to the Library-XML schema are:

- The vendor must supply an XML Schema fully describing their extended Library-XML format if it uses extended elements or attributes not already defined in the ODM namespace or Library-XML extension.

- Extended Library-XML files should reference the proper extension schema.
- The extension may add new XML elements and attributes, but may not render any standard Library-XML elements or attributes obsolete. Vendor extensions may not be used for information that is normally expressed using other Library-XML elements or attributes.
- Elements and attributes from the ODM and Define-XML schemas that are not a part of the Library-XML schema can be used, but no elements or attributes can be added to the ODM, Define-XML, or Library-XML namespaces.
- All extension elements and attributes not already defined in the ODM, Define-XML, or Library-XML namespaces must use a distinct XML namespace to ensure that there are no naming conflicts with other vendor extensions.
- The meaning of a Library-XML file must not be fundamentally changed by the addition of extensions.
- Removing all vendor extensions from an extended Library-XML file must result in a valid, meaningful and accurate Library-XML file.
- Vendors should be able to produce Library-XML files free of any vendor extensions upon request.

3.3 Library-XML Document Structure

The examples below show the basic structure of any ODM v1.3.2 or Define-XML v2.1 document that contains Library-XML content. The example is limited by design to highlight the basic structure of the XML. More detailed examples are provided in Section 4.

Example 3.3.1: Basic Library-XML document structure using Define-XML v2.1

```
<?xml version='1.0' encoding='utf8'?>
<ODM
    AsOfDateTime="2018-11-20T00:00:00+00:00" CreationDateTime="2021-06-22T17:06:00.974436+00:00"
    FileOID="ODM.SDTMIGv3.3.2018-11-20" FileType="Snapshot" Granularity="Metadata" ODMVersion="1.3.2"
    Originator="CDISC LIBRARY" SourceSystem="cdisc_library_api" SourceSystemVersion="1.0"
    def:Context="Other" mdr:LibraryXMLVersion="1.0.0"
    xmlns="http://www.cdisc.org/ns/odm/v1.3"
    xmlns:def="http://www.cdisc.org/ns/def/v2.1"
    xmlns:mdr="http://www.cdisc.org/ns/library-xml/v1.0"
    xmlns:xlink="http://www.w3.org/1999/xlink">
    <Study OID="STUDY.SDTMIG.v3.3">
        <GlobalVariables>
            <StudyName>SDTMIG v3.3</StudyName>
            <StudyDescription>Study Data Tabulation Model Implementation Guide Version 3.3</StudyDescription>
            <ProtocolName>CDISC Standard Publication</ProtocolName>
        </GlobalVariables>
        <MetaDataVersion
                Name="SDTMIG v3.3 Metadata" OID="MDV.SDTMIG.v3.3"
                def:DefineVersion="2.1.0" mdr:DatePublished="2018-11-20">
            <def:Standards>
                <def:Standard
                        Name="SDTMIG v3.3" OID="STD.SDTMIGv3.3" Status="Final" Type="IG" Version="3.3" />
            </def:Standards>
            <ItemGroupDef
                    OID="IGD.CO" Name="CO" Domain="CO" Repeating="No" Purpose="Tabulation"
                    def:Structure="One record per comment per subject">
                <Description>
                    <TranslatedText xml:lang="en">Comments</TranslatedText>
                </Description>
                <ItemRef
                        ItemOID="IT.CO.STUDYID" OrderNumber="1" Mandatory="No" Role="Identifier" />
                ...
                <def:Class Name="SPECIAL PURPOSE" />
            </ItemGroupDef>
            ...
            <ItemDef
                    DataType="text" Name="DOMAIN" OID="IT.CO.DOMAIN"
                    mdr:Core="Req" mdr:SubmissionDataType="Char"
                    mdr:DescribedValueDomain="ISO 8601" mdr:VariableSet="Identifier">
                <Description>
                    <TranslatedText xml:lang="en">Domain Abbreviation</TranslatedText>
                </Description>
                <CodeListRef CodeListOID="CL.C66734.CO" />
                <mdr:CDISCNotes>
                    <TranslatedText xml:lang="en">Two-character abbreviation for the domain.</TranslatedText>
                </mdr:CDISCNotes>
            </ItemDef>
            ...
            <CodeList
                    DataType="text" Name="CO Domain Abbreviation" OID="CL.C66734.CO">
                <EnumeratedItem CodedValue="CO" OrderNumber="1" />
                ...
                <Alias Context="nci:ExtCodeID" Name="C66734" />
            </CodeList>
            ...
            <CodeList DataType="text" Name="C66734" OID="CL.C66734">
                <ExternalCodeList
                        Dictionary="CDISC Library Controlled Terminology"
                        href="/mdr/root/ct/sdtmct/codelists/C66734" />
                <Alias Context="nci:ExtCodeID" Name="C66734" />
            </CodeList>
            ...
        </MetaDataVersion>
    </Study>
</ODM>
```

Example 3.3.2: Basic Library-XML document structure using ODM-XML v1.3.2

```
<?xml version='1.0' encoding='utf8'?>
<ODM
        AsOfDateTime="2019-11-01T00:00:00+00:00" CreationDateTime="2021-06-17T15:30:56.250720+00:00"
        FileOID="ODM.CDASHIGv2.1.2019-11-01" FileType="Snapshot"
        Granularity="Metadata" ODMVersion="1.3.2"
        Originator="CDISC LIBRARY" SourceSystem="cdisc_library_api" SourceSystemVersion="1.0"
        mdr:LibraryXMLVersion="1.0.0"
        xmlns="http://www.cdisc.org/ns/odm/v1.3"
        xmlns:mdr="http://www.cdisc.org/ns/library-xml/v1.0"
        xmlns:xlink="http://www.w3.org/1999/xlink">
    <Study OID="STUDY.CDASHIG.v2.1">
        <GlobalVariables>
            <StudyName>CDASHIG v2.1</StudyName>
            <StudyDescription>Clinical Data Acquisition Standards Harmonization IG</StudyDescription>
            <ProtocolName>CDISC Standard Publication</ProtocolName>
        </GlobalVariables>
        <MetaDataVersion
                Name="CDASHIG v2.1 Metadata" OID="MDV.CDASHIG.v2.1"
                mdr:DatePublished="2019-11-01" mdr:Status="Final">
            <ItemGroupDef
                    Domain="CO" Name="CO" OID="IGD.CO" Purpose="Data Collection" Repeating="No">
                <Description>
                    <TranslatedText xml:lang="en">Comments</TranslatedText>
                </Description>
                <ItemRef ItemOID="IT.CO.STUDYID" Mandatory="No" OrderNumber="1" />
                ...
                <mdr:Class Name="SPECIAL PURPOSE" />
            </ItemGroupDef>
            <ItemDef
                    DataType="text" Name="CMDOSU" OID="IT.CM.CMDOSU"
                    mdr:Core="R/C" mdr:SubmissionDataType="Char">
                <Description>
                    <TranslatedText xml:lang="en">CM Dose Units</TranslatedText>
                </Description>
                <Question>
                    <TranslatedText xml:lang="en">What is the unit for the dose?</TranslatedText>
                </Question>
                <CodeListRef CodeListOID="CL.C71620" />
                <Alias Context="sdtmigDatasetMappingTarget_0" Name="CM.CMDOSU" />
                <mdr:AltCodeListRef CodeListOID="CL.C78417" />
                <mdr:Prompt>
                    <TranslatedText xml:lang="en">(Dose) Unit</TranslatedText>
                </mdr:Prompt>
                <mdr:CRFCompletionInstructions>
                    <TranslatedText xml:lang="en">Record the dose unit of the dose taken.</TranslatedText>
                </mdr:CRFCompletionInstructions>
                <mdr:ImplementationNotes>
                    <TranslatedText xml:lang="en">When sponsors collect data for amount of dose
                        taken (i.e., Dose, Total Daily Dose), Unit must be collected as well.</TranslatedText>
                </mdr:ImplementationNotes>
                <mdr:MappingInstructions>
                    <TranslatedText xml:lang="en">Maps directly to the SDTMIG variable listed in the
                        column with the heading "SDTMIG Target".</TranslatedText>
                </mdr:MappingInstructions>
                <mdr:Definition>
                    <TranslatedText xml:lang="en">The unit associated with the concomitant
                        medication/treatment/therapy taken.</TranslatedText>
                </mdr:Definition>
            </ItemDef>
            ...
            <CodeList DataType="text" Name="C66781" OID="CL.C66781">
                <ExternalCodeList
                        Dictionary="CDISC Library Controlled Terminology"
                        href="/mdr/root/ct/sdtmct/codelists/C66781" />
                <Alias Context="nci:ExtCodeID" Name="C66781" />
            </CodeList>
            ...
        </MetaDataVersion>
    </Study>
</ODM>
```

3.4 Validation of a Library-XML document

A valid Library-XML document must:

- Properly reference versions of the CDISC standards.
- Be well formed and conform to the XML schemas.
- Meet all of the requirements documented in this specification.

Schema validation can only enforce some parts of the standards, so this additional level of validation is required to determine whether a Library-XML extended ODM or Define-XML document is fully compliant with the published specifications. See the XML Schema Validation for Define.xml White Paper for additional information.

The correct ordering of elements within a document is an absolute requirement for the document to be valid with respect to the Library-XML schema. The use of an XML schema definition and a validating parser environment makes detection of improperly ordered content fairly straightforward. In the absence of such mechanisms, care should be extended to following the order specified by the documentation for all extension content.

Note that XML is case sensitive, and case sensitivity plays a role in creating a valid Library-XML file. For example, ItemGroupOID=”DM” is not the same as ItemGroupOID=”dm”.

4 Using Library XML

The purpose of Library-XML is to enable CDISC Library and other MDRs to publish a broader set of metadata to describe the standards more completely. It supports the exchange of metadata between an MDR and software applications that manage or apply this metadata. The extension provides a common way to specify this content within ODM and Define-XML. The initial release of this specification contains a minimal set of additional metadata needed for the XML CDISC Library exports to match the content provided in the Excel exports. Future versions of this extension will add additional metadata as more metadata becomes available in CDISC Library, and more applications begin using this content.

The additional metadata provided in Define-XML through the Library-XML extension is not expected to be used as part of a regulatory submission. Library-XML is intended for use in metadata management scenarios.

4.1 Library-XML Extensions for CDASH

The current CDISC Library Exports Excel file for CDASH v1.1 contains several metadata elements not available in the basic ODM CDISC Library export. The Library-XML extension makes these available via ODM, as well as additional metadata elements requested by CDISC Library Archive users. For example, the ODM *ItemRef* element has a *Mandatory* attribute that takes a 'Yes' or 'No' value, whereas the CDASH Excel export has the CDASH *Core* attribute that takes values such as 'Highly Recommended'. The Library-XML extension adds an explicit attribute for *Core*. Library-XML also describes how to use existing ODM attributes to represent MDR content. For example, the *ItemGroupDef* attribute *Purpose* should be used for the CDISC Library Clinical Lifecycle Stage. The following table summarizes the content added to ODM to represent additional CDASH metadata.

Added CDASH Content	ODM Representation	ODM Values
Library-XML version	Add *LibraryXMLVersion* to *ODM*	1.0.0
Date published	Add a *DatePublished* attribute to *MetaDataVersion*	date e.g. Last Published date in CDISC Library
Lifecycle status	Add *Status* to *MetaDataVersion*	CDISC and CDISC Library lifecycle status e.g. Final, Provisional, Draft
Class	Add *Class* to *ItemGroupDef*	Observation classes for CDASH. Note: Class is not applicable to CDASH v1.1. e.g. EVENTS, FINDINGS, INTERVENTIONS, SPECIAL PURPOSE, TRIAL DESIGN

Core	Add *Core* attribute to *ItemDef*	(HR, R/C, O) HR = Highly Recommended R/C = Recommended/Conditional O = Optional
Datatype	Add *SubmissionDataType* attribute to *ItemDef*	(Char, Num)
Alternate CodeList	Add *AltCodeListRef* as a child element to *ItemDef*	e.g., <mdr:AltCodeListRef CodeListOID="CL.C78418"/>
Prompt	Add *Prompt* as a child element of *ItemDef*	text e.g., Visit Number
CRF Completion Instructions	Add *CRFCompletionInstructions* as a child element of *ItemDef*	text e.g., When applicable (e.g., on paper CRFs) record the visit number.
Implementation Notes	Add *ImplementationNotes* as a child element of *ItemDef*	text e.g., Rationale and implementation instructions
Mapping Instructions	Add *MappingInstructions* as a child element of *ItemDef*	text e.g., Information on suggested mapping
Definition	Add *Definition* as a child element of *ItemDef*	text e.g., Definition of the CDASHIG variable
Clinical Lifecycle Stage	Use the *ItemGroupDef* attribute *Purpose* for the lifecycle stage.	CDISC Library Clinical Lifecycle Stage e.g. Data Collection
Domain	Use the *ItemGroupDef* attribute *Domain* for the domain	Two character domain name e.g., AE, DM, VS
Question	Use the *Question* element as a child of *ItemDef* for the question	text e.g., What is the unit?
Maps to	Use the *Alias* element as a child of *ItemDef* to capture the variable in the "Mapping Target" relationship in CDISC Library.	e.g., <Alias Context="sdtmigDatasetMappingTarget_0" Name="CO.STUDYID"/>

CodeList items	Use the *ExternalCodeList* element to reference the code list	The code list items are not included with the code list for SDTM exports. The ExternalCodeList provides a reference to the code list and enables the XML to pass schema validation.
CodeList	Use the *Alias* element as a child of *CodeList* or *CodeListItem* to capture the NCI C-Code	e.g., <Alias Context="nci:ExtCodeID" Name="C78418">

4.1.1 Example of the Library-XML Extension used for CDASH

The example below corresponds to the AE ItemGroup and AEACN variable.

```
<ItemGroupDef Domain="AE" Name="AE" OID="IGD.AE" Purpose="Data Collection"
    Repeating="No">
    <Description>
        <TranslatedText xml:lang="en">Adverse Events</TranslatedText>
    </Description>
    <ItemRef ItemOID="IT.AE.AEACN" Mandatory="No" OrderNumber="34"/>
    ...
    <mdr:Class Name="EVENTS"/>
</ItemGroupDef>

<ItemDef DataType="text" Name="AEACN" OID="IT.AE.AEACN" mdr:Core="R/C"
    mdr:SubmissionDataType="Char">
    <Description>
        <TranslatedText xml:lang="en">Action Taken with Study Treatment</TranslatedText>
    </Description>
    <Question>
        <TranslatedText xml:lang="en">What action was taken with...</TranslatedText>
    </Question>
    <CodeListRef CodeListOID="CL.C66767"/>
    <Alias Context="sdtmigDatasetMappingTarget_0" Name="AE.AEACN"/>
    <mdr:Prompt>
        <TranslatedText xml:lang="en">Action Taken with Study Treatment</TranslatedText>
    </mdr:Prompt>
    <mdr:CRFCompletionInstructions>
        <TranslatedText xml:lang="en">Record changes made to the...</TranslatedText>
    </mdr:CRFCompletionInstructions>
    <mdr:ImplementationNotes>
        <TranslatedText xml:lang="en">CDISC Controlled Terminology is...</TranslatedText>
    </mdr:ImplementationNotes>
    <mdr:MappingInstructions>
        <TranslatedText xml:lang="en">Maps directly to the SDTMIG...</TranslatedText>
    </mdr:MappingInstructions>
    <mdr:Definition>
        <TranslatedText xml:lang="en">A description of the action...</TranslatedText>
    </mdr:Definition>
</ItemDef>
```

4.2 Library-XML Extensions for ADaM, SDTM and SEND

The current CDISC Library Archive Excel file for SDTM v3.2 contains several metadata elements not available in the Define-XML SDTM CDISC Library export. The Library-XML extension makes this metadata available via Define-XML. It also adds additional metadata elements requested by CDISC Library Archive users. For example the Define-XML *ItemRef* element has a *Mandatory* attribute that takes a 'Yes' or 'No' value, whereas the SDTM Excel export has the SDTM *Core* attribute that takes values such as 'Required'. The Library-XML extension adds an explicit attribute for *Core*.

Not every element and attribute of the Define-XML v2.1 extension is supported by Library-XML since not all are appropriate for MDR content. For example, *def:ArchiveLocationID* is not applicable in the Library-XML use case. The following Define-XML elements and attributes are supported:

- *def:Context*
- *def:DefineVersion*
- *def:Standards*
- *def:Structure*
- *def:Class*

The following table summarizes the content added to Define-XML to represent additional SDTM metadata. Many of the elements and attributes added via the extension are the same for both SDTM and CDASH content (e.g., *Core*).

Added SDTM Content	**ODM Representation**	**ODM Values**
Library-XML version	Add *LibraryXMLVersion* to *ODM*	1.0.0
Context	Use *def:Context* in *ODM* for define-XML content	(Submission, Other)
Date published	Add a *DatePublished* attribute to *MetaDataVersion*	date e.g. Last Published date in CDISC Library
Define version	Add a *def:DefineVersion* attribute to *MetaDataVersion*	2.1.0
Standards	Add *def:Standards* as a child element of *MetaDataVersion* for define-XML content	text e.g. <def:Standards> <def:Standard OID="STD.SDTMIGv3.3" Name="SDTMIG v3.3" Version="3.3" Type="IG" Status="Final" /> </def:Standards>

Structure	Use *def:Structure* as part of *ItemGroupDef* for Define-XML content	text Description of the level of detail represented by individual records in the dataset, such as "One record per medical history event per subject" for the MH domain
Class	Use *def:Class* in *ItemGroupDef* for Define-XML content	Observation classes for SDTM e.g. Events, Findings, Interventions, Special Purpose, Trial Design, Relationship
Core	Add *Core* attribute to *ItemDef*	(Req, Exp, Perm, Cond)
Datatype	Add *SubmissionDataType* attribute to *ItemDef*	(Char, Num)
Variable Set	Add *VariableSet* attribute to *ItemDef*	text ADaM variable set, e.g. Dose, Analysis Parameter, Treatment Timing
Described Value Domain	Add a *DescribedValueDomain* attribute to *ItemDef*	The format used to represent the data. In CDISC Library format exists as a Value Domain classifier. e.g., ISO 8601
Alternate CodeList	Add *AltCodeListRef* as a child element to *ItemDef*	e.g., <mdr:AltCodeListRef CodeListOID="CL.C66727"/>
CDISC Notes	Add *CDISCNotes* as a child element of *ItemDef*	text e.g., Clinical encounter number. Numeric version of VISIT, used for sorting.
Clinical Lifecycle Stage	Use the *ItemGroupDef* attribute *Purpose* for the lifecycle stage.	CDISC Library Clinical Lifecycle Stage e.g. Data Tabulations
Domain	Use the *ItemGroupDef* attribute *Domain* for the domain	Two character domain name e.g., AE, DM, VS
CodeList C-Code	Use the *Alias* element as a child of *CodeList* or *CodeListItem* to capture the NCI C-Code	e.g., <Alias Context="nci:ExtCodeID" Name="C66728">
Code list items	Use the *ExternalCodeList* element to reference the code list	The code list items are not included with the code list for SDTM exports. The *ExternalCodeList* provides a reference to the code list and enables the XML to pass schema validation.

4.2.1 Example of the Library-XML Extension used for SDTM

The example below corresponds to the AE ItemGroup and AEACN variable.

```
<ItemGroupDef Domain="AE" Name="AE" OID="IGD.AE" Purpose="Tabulation" Repeating="No"
    def:Structure="One record per adverse event per subject">
    <Description>
        <TranslatedText xml:lang="en">Adverse Events</TranslatedText>
    </Description>
    <ItemRef ItemOID="IT.AE.AEACN" Mandatory="No" OrderNumber="28" Role="Record Qualifier"/>
    <def:Class Name="EVENTS"/>
</ItemGroupDef>

<ItemDef DataType="text" Name="AEACN" OID="IT.AE.AEACN"
    mdr:Core="Exp" mdr:SubmissionDataType="Char">
    <Description>
        <TranslatedText xml:lang="en">Action Taken with Study Treatment</TranslatedText>
    </Description>
    <CodeListRef CodeListOID="CL.C66767"/>
    <mdr:CDISCNotes>
        <TranslatedText xml:lang="en">Describes changes to the study...</TranslatedText>
    </mdr:CDISCNotes>
</ItemDef>

<CodeList DataType="text" Name="C66767" OID="CL.C66767">
    <ExternalCodeList Dictionary="CDISC Library Controlled Terminology"
        href="/mdr/root/ct/sdtmct/codelists/C66767"/>
    <Alias Context="nci:ExtCodeID" Name="C66767"/>
</CodeList>
```

4.3 Library-XML Extensions for API Support

In addition to the MDR content extensions Library-XML adds information to make it easier to work with the API. The use of the *ExternalCodeList* element has been recommended for cases when *CodeLists* are included to support variable definitions without including all the terms and in this case the *href* attribute provides the web services URL to retrieve the *CodeList's* terms.

4.3.1 Example of the ExternalCodeList Use with API Support

The example below shows the expected use of the *ExternalCodeList* element.

```
<ExternalCodeList Dictionary="ACN"
    href="https://smpoc3.akana.com/sm/rest/semantics/CDISCAPI/
    standards/1.0_13909674146771678100 26/metadata"/>
```

4.4 Library-XML use of Alias

In addition to extended content, Library-XML prescribes the use of certain elements to carry additional metadata. The *Alias* element is prescribed to provide the NCI/EVS C-Code for any *CodeList* or *CodeListItem* referenced in the metadata. The use of the *ExternalCodeList* element as a reference to CodeList terms was mentioned in the previous section.

4.4.1 Example of the Alias as a child element of CodeList

The example below shows the expected use of the *Alias* in to supplement *CodeList* and *CodeListItem* with the NCI/EVS C-Code.

```
<Alias Context="nci:ExtCodeID" Name="C66767"/>
```

4.4.2 Example of the Alias as a child element of ItemDef

Alias is used as a child element of *ItemDef* when additional context is available in the MDR. This additional context content is included when available, and may not exist for all variables. Maps-to relationship exist for many variables in CDISC Library to show traceability through the clinical research data lifecycle. Note that *Alias Context* attribute values must be unique per *ItemDef*. The example below shows the expected use of *Alias* to supplement *ItemDef* for

```
<Alias Context="sdtmigDatasetMappingTarget_0" Name="EC.ECDOSE"/>
<Alias Context="sdtmigDatasetMappingTarget_1" Name="EC.ECDOSTXT"/>
```

CDASH content.

5 Specification

5.1 Library-XML Scope

Library-XML represents standards metadata for ADaM, CDASH, SDTM, and SEND. Library-XML will also be used to represent the TAUG content once these formats have been finalized. It extends the metadata content available in ODM and Define-XML to represent metadata provided by an MDR or other metadata library. The Library-XML extension will be used by CDISC Library as the format for providing access to metadata in XML via the Library-API.

5.2 Library-XML Structure

Library-XML is based on the CDISC ODM standard, and extends the existing ODM and Define-XML structures. A Library-XML file includes all the components needed to create a valid ODM or Define-XML document. Library-XML represents only metadata, and elements representing data are not included. For CDASH, Library-XML works with ODM v1.3.2 and later. For ADaM, SDTM and SEND, Library-XML works with Define-XML v2.1 and later.

The sections that follow describe the content of the Library-XML extension and the elements that contain them. This document is intended to complement the ODM and Define-XML specifications, and does not specify all the elements needed to create a valid Library-XML document. An XML document that implements the Library-XML extension should be a valid ODM or Define-XML document prior to the Library-XML content being added. Conversely, if the Library-XML extended elements and attributes are removed from the document, a valid ODM or Define-XML document should remain.

Each of the following sections describes the elements in the order in which they occur in the XML. Elements that may be used in more than one context are presented where they first appear in the document. The organization of the elements in this specification document does not reflect the hierarchical XML structure. The ODM or Define-XML specifications are needed to fill the gaps in the XML specified in this document.

Each section begins with a brief description of the element followed by an *element table*, and an *attribute table*. The examples are provided in Section 4 General Specifications for Library-XML.

An *element table* describes the different aspects of an element's definition while the *attribute table* describes the element's attributes. The following templates illustrate the layouts of these tables, including the header and a description of the content.

Element Table Template

Element Name:	*Name of the element*
Element XPath:	*XPath showing where the element belongs in the XML*
Element Textual Value:	*A description of the value of the element. If an element has no text value (e.g. it has child elements instead), then this cell is populated with "None".*
Usage:	Requirement: *This is populated with one of three values:* • *"Required" when at least one instance of the element is required* • *"Optional" when the element is optional* • *"Conditional" when at least one instance of the element is required under certain conditions. It will include the conditions under which the element is Required.* Cardinality: *This indicates the number of instances expected (e.g.* "Exactly One", "One or more", *etc.)* Business Rule(s): *This is populated with rules that have to be satisfied in addition to an XML schema validation for a Library-XML document to be considered compliant with the Library-XML v1.0.0 specification.* Other Information: *This is populated with any other information about the element, including the conditions under which the element is included, how the schema is applied to support the model, relative position of the element in the model, etc*.
Attributes:	*A comma-delimited list of the attributes of this element. If the element has no attributes, this is populated with* "None".
Child Elements:	*A comma-delimited list of the immediate child elements of this element. If the element has no child elements, this is populated with "None". The order of child elements shown in the specification is the order in which they must appear in an Library-XML document.* *A link to a child element will be provided when the child element is described in a different section of the document and not under a sub-section of the element being described or in the section or subsection immediately following the current element.*

Attribute Table Template

Attribute	Usage	Allowable Values	Description
Name of the attribute	*This is populated with* "Required" *when the attribute is required,* "Optional" *when the attribute is optional, or* "Conditional" *when the attribute is required under certain conditions.* *It will include the conditions under which the attribute is Required.* *Default: This will be populated with a default value if one is provided in the specification.*	*Any combination of the following:* Allowable Value: *The only allowed value* Allowable Values: *A comma-delimited list of the allowable values* Value Description: *A textual description of allowable values* See Appendix *xx:* *A reference to an appendix including a hyperlink to the appendix* Sample: *An example*	*A textual description of the attribute beyond what is included in the Allowable Values column.* Business Rule(s): *Rules that have to be satisfied in addition to schema validation for an Library-XML document to be considered compliant with the Library-XML v1.0.0 specification.*

5.3 Library-XML Specification Details

The Library-XML specification details describe the content of the Library-XML extension and the elements that contain them. This document is intended to complement the ODM and Define-XML specifications.

5.3.1 XML Header

The first line of a Library-XML file must be the XML header. The XML header indicates that the remainder of the file is XML and specifies the character encoding it uses.

Example XML Header

```
<?xml version="1.0" encoding="UTF-8"?>
```

This example shows a Library-XML file using the "UTF-8" character encoding.

5.3.2 ODM Element

The first XML element in a file is known as the root element. In Library-XML the *ODM* element is the root element. The *ODM* element identifies the namespaces used, and includes attributes that affect the processing of the document as a whole.

Element Name:	ODM
Element XPath:	/ODM
Element Textual Value:	*None*
Usage:	Requirement: Required Cardinality: Exactly one Other Information: This is the root element for the Library-XML document
Attributes:	xmlns, **xmlns:mdr**, **xmlns:def**, xmlns:xlink, ODMVersion, FileType, Granularity, FileOID, CreationDateTime, AsOfDateTime, Originator, SourceSystem, SourceSystemVersion, def:Context, **mdr:LibraryXMLVersion**
Child Elements:	Study

Attribute	Usage	Allowable Values	Description
xmlns	Required	"http:// www.cdisc.org/ ns/odm/v1.3"	Identifies the default namespace for this document.
xmlns:mdr	Required	"http:// www.cdisc.org/ns/ library-xml/v1.0"	XML namespace for Library-XML v1.0.0. While "mdr:" is the suggested prefix for the Library-XML namespace, it should not be relied upon by the receiving application. The namespace prefix should follow the W3C naming requirements.

xmlns:def	Conditional Required when extending Defin-XML.	"xmlns:def="http://www.cdisc.org/ns/def/v2.1"	XML namespace for Define-XML v2.1.0. While "def:" is the suggested prefix, it should not be relied upon by the receiving application. The namespace prefix should follow the W3C naming requirements.
xmlns:xlink	Conditional Required when xlink:href is provided.	"http://www.w3.org/1999/xlink"	XML namespace for XLink.
ODMVersion	Required	"1.3.2"	Identifies the ODM version that underlies the schema for the Library-XML document. ODMVersion is optional in the ODM standard, but required in Library-XML.
FileType	Required	"Snapshot"	Library-XML documents do not include audit trail elements, so the FileType is Snapshot .
Granularity	Optional	"Metadata"	A Granularity of "Metadata" should be used for Library-XML files.
FileOID	Required	Text	A unique identifier for this file. See the ODM specification for a discussion of FileOID recommendations.
CreationDateTime	Required	ISO8601 datetime Sample: "2013-09-30T15:31:04"	The date and time when the specific version of the Library-XML file was created. This is the "last modified" date and time.
AsOfDateTime	Optional	ISO8601 datetime Sample: "2013-09-30T15:31:04"	The date and time at which the source database was queried to create this document.
Originator	Optional	Text Sample: "Company XYZ"	Submission sponsor name.
SourceSystem	Optional	Text	The name of the application that generated the Library-XML file.
SourceSystemVersion	Optional	Text	The version of the "SourceSystem" above.
def:Context	Conditional Required when extending Defin-XML.	Allowable Values: Submission, Other	Context in which the Library-XML document is used.
mdr:LibraryXMLVersion	Required	"1.0.0"	Library-XML version identification.

5.3.3 MetaDataVersion Element

The *MetaDataVersion* element contains all the definitions related to the metadata extracted from the MDR (e.g. CDISC Library) in the Library-XML document. It can include attributes to identify the versions of standards like CDISC SDTM and Define-XML when extending *Define-XML*.

The table below specifies how the *MetadataVersion* element in a Library-XML file shall be constructed.

Element Name:	MetaDataVersion
Element XPath:	/ODM/Study/MetaDataVersion
Element Textual Value:	*None*
Usage	Requirement: Required Cardinality: One Only one MetaDataVersion element is supported for use in Library-XML.
Attributes:	OID, Name, def:DefineVersion, **mdr:DatePublished**, **mdr:Status**
Child Elements:	def:Standards, ItemGroupDef, ItemDef, CodeList

Attribute	Usage	Allowable Values	Description
OID	Required	Text Sample: "MDV.SDTM.2014-07-15_13:27:24", "MDV.CDASH.2014-01-31_12:07:24"	Unique ID for the MetaDataVersion. See the ODM specification section 2.11 for OID considerations.
Name	Required	Text	Name for the MetaDataVersion.
def:DefineVersion	Conditional Required when extending Define-XML	Text Sample: "2.1.0"	Version of Define-XML that the file conforms to. Used for the SDTM, SEND, and ADaM standards.
mdr:DatePublished	Required	ISO8601 date Sample: "2013-09-30"	Date last published within the MDR (e.g. Last Published date in CDISC Library)
mdr:Status	Conditional Required when extending ODM	Allowable Values: Draft, Provisional, Final	Lifecycle or governance status (e.g. lifecycle status within CDISC Library)

5.3.3.1 Example MetaDataVersion Element

The following XML shows an example Library-XML *MetaDataVersion* element for SDTMIG 3.1.2 represented in Define-XML v2.1.0.

```
<MetaDataVersion
  OID="CDISC01.SDTMIG.3.1.2.SDTM.1.2"
  Name="SDTM MDV"
  def:DefineVersion="2.1.0"
  mdr:DatePublished="2013-09-30">
```

5.3.3.2 Alias Element

Library-XML does not extend the *Alias* element but does prescribe its use to represent additional metadata. Library-XML uses the *Alias* element to provide a reference to external mappings for CDISC Terminology NCI/EVS C-Codes. CDISC Library also adds contextual content to *ItemDef* when available including Maps-to relationships. See section 4.4 for examples on the use of *Alias* in Library-XML.

Element Name:	Alias
Element XPath(s):	/ODM/Study/MetaDataVersion/ItemDef/Alias /ODM/Study/MetaDataVersion/CodeList/Alias /ODM/Study/MetaDataVersion/CodeList/CodeListItem/Alias /ODM/Study/MetaDataVersion/CodeList/EnumeratedItem/Alias
Element Textual Value:	*None*
Usage	Requirement: Conditional Cardinality: One or more • When used for Library-XML Alias is required as the means to provide the C-Code for CodeList, CodeListItem, and EnumeratedItem.
Attributes:	Context, Name
Child Elements:	*None*

Attribute	Usage	Allowable Values	Description
Context	Required	Text Sample: • Context for *Alias* as a child element of *ItemDef*: sdtmigDatasetMappingTarget_<n> • Context for *Alias* as a child element of a CodeList, CodeListItem or EnumeratedItem element: nci:ExtCodeID	Indicates the context or setting where the Alias Name attribute applies.
Name	Required	Text Sample: • Name when Context="sdtmigDatasetMappingTarget_<n>": name of the variable mapped to in CDISC Library. • Name when Context="nci:ExtCodeID": C-Code for corresponding CDISC Controlled Terminology codelist or codelist item.	Alternative Name for parent element.

5.3.4 ItemGroupDef Element

The *ItemGroupDef* element with the set of attributes and child elements as described below is used to describe the case report form item groups or dataset metadata in XML.

Element Name:	ItemGroupDef

Element XPath:	/ODM/Study/MetaDataVersion/ItemGroupDef
Element Textual Value:	*None*
Usage	Requirement: Required Cardinality: One or more
Attributes:	OID, Name, Repeating, Domain, Purpose, def:Structure
Child Elements:	Description, ItemRef, def:Class, **mdr:Class**

Attribute	Usage	Allowable Values	Description
OID	Required	Text	Unique ID for the ItemGroupDef. See the ODM specification section 2.11 for OID considerations.
Name	Required	Text	Short description for the ItemGroup.
Domain	Conditional • Required for SDTM and CDASH content.	Text	Two character domain designation
Repeating	Required	Allowable Values: Yes, No	Indicates whether a domain contains more than one record per subject or only one record per subject.
Purpose	Optional	Samples: Data Collection, Tabulation, Analysis.	Purpose of domain or dataset (e.g. Clinical Lifecycle Stage in CDISC Library)
def:Structure	Optional • Used with Define-XML only	text Samples: MH domain: "One record per medical history event per subject" VS domain: "One record per vital sign measurement per visit per subject"	Description of the level of detail represented by individual records in the dataset.

5.3.4.1 mdr:Class

Library-XML adds the *mdr:Class* child element to the *ItemGroupDef* element to represent the General class of the data domain.

Element Name:	mdr:Class
Element XPath:	/ODM/Study/MetaDataVersion/ItemGroupDef/mdr:Class
Element Textual Value:	*None*
Usage	Requirement: Conditional

	Cardinality: • One class per ItemGroupDef when using Library-XML and ODM. • def:Class is used instead when working with Define-XML
Attributes:	Name
Child Elements:	None

Attribute	Usage	Allowable Values	Description
Name	Required	text Samples: CDASH: SPECIAL PURPOSE FINDINGS EVENTS INTERVENTIONS TRIAL DESIGN RELATIONSHIP	General class of the data domain.

5.3.5 ItemDef Element

Library-XML extends the *ItemDef* element to include additional attributes and child elements to represent additional metadata.

Element Name:	ItemDef
Element XPath:	/ODM/Study/MetaDataVersion/ItemDef
Element Textual Value:	*None*
Usage	Requirement: Required Cardinality: • An ItemDef element is required for each ItemOID value that appears in an ItemRef contained in a MetaDataVersion.
Attributes:	OID, Name, DataType, **mdr:Core**, **mdr:SubmissionDataType, mdr:DescribedValueDomain, mdr:VariableSet**
Child Elements:	Description, Question, CodeListRef, Alias, **mdr:AltCodeListRef**, **mdr:Prompt**, **mdr:CRFCompletionInstructions**, **mdr:ImplementationNotes**, **mdr:MappingInstructions**, **mdr:Definition**, **mdr:CDISCNotes**

Attribute	Usage	Allowable Values	Description
OID	Required	text	Unique ID for the ItemDef (variable/value). See the ODM specification section 2.11 for OID considerations.
Name	Required	text	Variable name.
DataType	Required	Allowable Values: Refer to the ODM or Define-XML specification for details on valid data types.	The data type of the variable or value.

mdr:Core	Conditional	text Sample: CDASH: HR, R/C, O; SDTM: Req, Exp, Perm	SDTM and CDASH Core designations. Reference the appropriate specifications for more a complete description.
mdr:SubmissionDataType	Required	Text Allowable Values: Char, Num	The SDTM data type used to represent the ItemDef's value.
mdr:DescribedValueDomain	Optional	text Sample: ISO 8601	The SDTM described value domain or format
mdr:VariableSet	Conditional • Required for ADaM content.	text Sample: Dose, Analysis Parameter, Treatment Timing	ADaM variable set

Note: Length metadata is not available in CDISC Library, thus the Length and SignificantDigits attributes are not typically used in Library-XML. The Define-XML extension attributes *def:DisplayFormat* and *def:CommentOID* are not currently used by Library-XML. Attributes not listed here, but valid for ODM or Define-XML, are permissible for use in Library-XML.

5.3.5.1 mdr:AltCodeListRef

Library-XML adds the *mdr:AltCodeListRef* child element to the *ItemDef* element to represent the alternate codelists for item values.

Element Name:	mdr:AltCodeListRef
Element XPath:	/ODM/Study/MetaDataVersion/ItemDef/mdr:AltCodeListRef
Element Textual Value:	*Alternate codelist for item values.*
Usage	Requirement: Conditional Cardinality: • One AltCodeListRef per alternate codelist
Attributes:	CodeListOID
Child Elements:	None

Attribute	Usage	Allowable Values	Description
CodeListOID	Required	Text	Unique ID for the CodeList. See the ODM specification section 2.11 for OID considerations.

5.3.5.2 mdr:Prompt

Library-XML adds the *mdr:Prompt* child element to the *ItemDef* element to represent the CDASH Prompt content.

Element Name:	mdr:Prompt
Element XPath:	/ODM/Study/MetaDataVersion/ItemDef/mdr:Prompt
Element Textual Value:	*The CDASH short prompt or label for the data collection field.*
Usage	Requirement: Conditional Cardinality: • One prompt per ItemDef where a prompt exists.
Attributes:	None
Child Elements:	TranslatedText

5.3.5.3 mdr:CRFCompletionInstructions

Library-XML adds the *mdr:CRFCompletionInstructions* child element to the *ItemDef* element to represent the CDASH CRF Completion Instructions content.

Element Name:	mdr:CRFCompletionInstructions
Element XPath:	/ODM/Study/MetaDataVersion/ItemDef/mdr:CRFCompletionInstructions
Element Textual Value:	*The CDASH information for the clinical site on how to enter collected information on the CR.*
Usage	Requirement: Conditional Cardinality: • One "CRF Completion Instructions" element per ItemDef where such content exists.
Attributes:	None
Child Elements:	TranslatedText

5.3.5.4 mdr:ImplementationNotes

Library-XML adds the *mdr:ImplementationNotes* child element to the *ItemDef* element to represent the CDASH Implementation Notes content.

Element Name:	mdr:ImplementationNotes
Element XPath:	/ODM/Study/MetaDataVersion/ItemDef/mdr:ImplementationNotes
Element Textual Value:	*Further information, such as rationale and implementation instructions, on how to implement the CRF data collection fields and how to map CDASHIG variables to SDTMIG variables.*
Usage	Requirement: Conditional Cardinality: • One "Implementation Notes" element per ItemDef where such content exists.
Attributes:	None
Child Elements:	TranslatedText

5.3.5.5 mdr:MappingInstructions

Library-XML adds the *mdr:MappingInstructions* child element to the *ItemDef* element to represent the CDASH Mapping Instructions content.

Element Name:	mdr:MappingInstructions

Element XPath:	/ODM/Study/MetaDataVersion/ItemDef/mdr:MappingInstructions
Element Textual Value:	*Information on the suggested mapping of the CDASHIG variable to the SDTMIG variable.*
Usage	Requirement: Conditional Cardinality: • One "Mapping Instructions" element per ItemDef where such content exists.
Attributes:	None
Child Elements:	TranslatedText

5.3.5.6 mdr:Definition

Library-XML adds the *mdr:Definition* child element to the *ItemDef* element to represent the CDASH Definition content.

Element Name:	mdr:Definition
Element XPath:	/ODM/Study/MetaDataVersion/ItemDef/mdr:Definition
Element Textual Value:	*Draft definition of the CDASHIG variable.*
Usage	Requirement: Conditional Cardinality: • One "Definition" element per ItemDef where such content exists.
Attributes:	None
Child Elements:	TranslatedText

5.3.5.7 mdr:CDISCNotes

Library-XML adds the *mdr:CDISCNotes* child element to the *ItemDef* element to represent the CDISC Notes content.

Element Name:	mdr:CDISCNotes
Element XPath:	/ODM/Study/MetaDataVersion/ItemDef/mdr:CDISCNotes
Element Textual Value:	*Content that contains notes to the sponsor regarding the relevant to the use of each variable.*
Usage	Requirement: Conditional Cardinality: • One "CDISC Notes" element per ItemDef where such content exists.
Attributes:	None
Child Elements:	TranslatedText

5.3.6 CodeList Element

The *CodeList* element is not extended in Library-XML, but the use of Alias is prescribed to represent the NCI/EVS C-Codes when the CDISC Controlled Terminology is used. The use of the ExternalCodeList child element is also prescribed for cases when referring to the CDISC Terminology terms for a CodeList is preferred to listing each term in the metadata.

Element Name:	CodeList
Element XPath:	/ODM/Study/MetaDataVersion/CodeList
Element Textual Value:	*None*
Usage	Requirement: Conditional Cardinality: A CodeList element must be provided for each distinct value of the CodelistOID attribute in a CodeListRef element in the MetaDataVersion.
Attributes:	OID, Name, DataType
Child Elements:	EnumeratedItem, CodeListItem, ExternalCodeList, Alias

Attribute	Usage	Allowable Values	Description
OID	Required	Text	Unique ID for the CodeList. See the ODM specification section 2.11 for OID considerations.
Name	Required	Text	Controlled Terminology name. Business Rule: For NCI/CDISC Controlled Terminology, this must exactly match the CodeList Name from the published Controlled Terminology ODM.
DataType	Required	Allowable Values: text float integer	The data type of the codes.

5.3.6.1 ExternalCodeList

Library-XML uses the ExternalCodeList to represent CDISC controlled terminology when it is not included in the Library-XML document content. In this instance, ExternalCodeList represents CDISC controlled terminology created and maintained by the NCI/EVS and also available in CDISC Library. ExternalCodeList is not extended for Library-XML but a specific usage is prescribed.

Element Name:	ExternalCodeList
Element XPath:	/ODM/Study/MetaDataVersion/CodeList/ExternalCodeList
Element Textual Value:	None
Usage	Requirement: Conditional • Required when the sender prefers to reference the terms for a CDISC Controlled Terminology CodeList rather than listing each term in the metadata. In this casea n ExternalCodeList element must be provided to identify the Name and Version of the terminology. A reference to the terms is also desireable. Cardinality: One
Attributes:	Dictionary, Version, ref, href

Child Elements:	None

Attribute	Usage	Allowable Values	Description
Dictionary	Required	Text Sample: "ACN", "CMDOSFRM"	For Library-XML the CDISC submission value for the codelist is recommended since CodeList carries the full codelist name.
Version	Optional	Text Sample: "2014-06-27"	For Library-XML this is not currently used.
ref	Optional	Text	Reference to a local instance of the dictionary.
href	Optional	Text	URL to retrieve this codelist resource using the CDISC Library API.

6 Acknowledgments

This specification was developed for CDISC by

- Sam Hume, CDISC.

Input to this specification was provided by the CDISC Data Science and Data Exchange Standards teams, and special thanks go to:

- Anthony Chow, CDISC
- Sally Cassells, CDISC
- Marcelina Hungria, DIcore Group, LLC
- Jozef Aerts, XML4Pharma

7 Appendices

7.1 Appendix A: XML Schema

These XML files reference (directly or indirectly) the following schema files:

Library-XML Schema Files
schema/cdisc-define-2.1/define2-1-0.xsd
schema/cdisc-define-2.1/define-enumerations.xsd
schema/cdisc-define-2.1/define-extension.xsd
schema/cdisc-define-2.1/define-ns.xsd
schema/cdisc-odm-1.3.2/ODM1-3-2.xsd
schema/cdisc-odm-1.3.2/ODM1-3-2-foundation.xsd
schema/core/xlink.xsd
schema/core/xml.xsd
schema/core/xmldsig-core-schema.xsd
schema/library-xml-1.0.0/library-def-1-0-0.xsd
schema/library-xml-1.0.0/library-def-enumerations.xsd
schema/library-xml-1.0.0/library-def-extension.xsd
schema/library-xml-1.0.0/library-enumerations.xsd
schema/library-xml-1.0.0/library-ns.xsd
schema/library-xml-1.0.0/library-odm-1-0-0.xsd

schema/library-xml-1.0.0/library-odm-enumerations.xsd
schema/library-xml-1.0.0/library-odm-extension.xsd

7.2 Appendix B: Library-XML Programming Specifications

This document's purpose is to consolidate instructions in SHARE20QC-411.

Instructions for specific XML attribute:

XML Attribute	Standards	Logic
Repeating	All (CDASH, SDTM, ADaM, and their IG)	Default value is Repeating="Yes", except when datasets or domains in these names, in which case Repeating="No": • ADSL • APDM • DI • DM • DO • DT • OI • PB • TA • TD • TE • TI • TM • TP • TS • TT • TV • TX
Mandatory	All (CDASH, SDTM, ADaM, and their IG)	Default value is Mandatory="No", except: • For SDTMIG, SENDIG, or ADaMIG, when "Core=Req", set Mandatory="Yes" • For CDASHIG, when "Core= H/R", set Mandatory="Yes"
DataType	All (CDASH, SDTM, ADaM, and their IG)	Defaults to the following: if mdr:SubmissionDataType="Char", then DataType="text" else if mdr:SubmissionDataType="Num" then DataType="float" except for variables in these names, plus those instantiated to domains & datasets in the IG using library-std:implementsVariable, For example, --DTC would mean --DTC in Model and LBDTC, MBDTC, etc. in IG: variable name | type

XML Attribute	Standards	Logic	
		--DTC	datetime
		--STDTC	datetime
		--ENDTC	datetime
		--RFTDTC	datetime
		MIDSDTC	datetime
		--DUR	durationDatetime
		--EVLINT	durationDatetime
		--ELTM	durationDatetime
		--STINT	durationDatetime
		--ENINT	durationDatetime
		TDSTOFF	durationDatetime
		TDTGTPAI	durationDatetime
		TDMINPAI	durationDatetime
		TDMAXPAI	durationDatetime
		--LLTCD	integer
		--PTCD	integer
		--HLTCD	integer
		--HLGTCD	integer
		--SOCCD	integer
		--DY	integer
		VISITDY	integer
		AGE	integer
		--STDY	integer
		--ENDY	integer
		--DETECT	integer
		--BDSYCD	integer
		--REPNUM	integer
		--RPPLDY	integer
		--NOMDY	integer
		--RPDY	integer
		--RPSTDY	integer
		--RPENDY	integer
		EGBEATNO	integer
		TDNUMRPT	integer

Other handling instructions:

FileOID	"library_cdisc_org" + Study OID or endpoint (change / to _). E.g.: • library_cdisc_org_mdr_cdashig_2-0 • library_cdisc_org_mdr_sdtmig_3-2

OID values	Use the endpoint listed in the current examples, but replace the / with . E.g.: • OID="mdr.cdashig.2-0.domains.CM" • ItemOID="mdr.sdtmig.3-2.datasets.CO.variables.STUDYID"
mdr:Status and mdr:DatePublished	Since we're using roots, we can drop the mdr:Status and mdr:DatePublished from CodeList. Use ExternalCodeList to represent the endpoint so that the root contents can be retrieved. Use Alias to provide the c-code. We did this for SHARE 1.0 as well.

7.3 Appendix C: Representations and Warranties, Limitations of Liability, and Disclaimers

CDISC Patent Disclaimers
It is possible that implementation of and compliance with this specification may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any claim or of any patent rights in connection therewith. CDISC, including the CDISC Board of Directors, shall not be responsible for identifying patent claims for which a license may be required in order to implement this standard or for conducting inquiries into the legal validity or scope of those patents or patent claims that are brought to its attention.

Representations and Warranties
"CDISC grants open public use of this Specification (or Final Standards) under CDISC's copyright."

Each Participant in the development of this standard shall be deemed to represent, warrant, and covenant, at the time of a Contribution by such Participant (or by its Representative), that to the best of its knowledge and ability: (a) it holds or has the right to grant all relevant licenses to any of its Contributions in all jurisdictions or territories in which it holds relevant intellectual property rights; (b) there are no limits to the Participant's ability to make the grants, acknowledgments, and agreements herein; and (c) the Contribution does not subject any Contribution, Draft Standard, Final Standard, or implementations thereof, in whole or in part, to licensing obligations with additional restrictions or requirements inconsistent with those set forth in this Policy, or that would require any such Contribution, Final Standard, or implementation, in whole or in part, to be either: (i) disclosed or distributed in source code form; (ii) licensed for the purpose of making derivative works (other than as set forth in Section 4.2 of the CDISC Intellectual Property Policy ("the Policy")); or (iii) distributed at no charge, except as set forth in Sections 3, 5.1, and 4.2 of the Policy. If a Participant has knowledge that a Contribution made by any Participant or any other party may subject any Contribution, Draft Standard, Final Standard, or implementation, in whole or in part, to one or more of the licensing obligations listed in Section 9.3, such Participant shall give prompt notice of the same to the CDISC President who shall promptly notify all Participants.

No Other Warranties/Disclaimers. ALL PARTICIPANTS ACKNOWLEDGE THAT, EXCEPT AS PROVIDED UNDER SECTION 9.3 OF THE CDISC INTELLECTUAL PROPERTY POLICY, ALL DRAFT STANDARDS AND FINAL STANDARDS, AND ALL CONTRIBUTIONS TO FINAL STANDARDS AND DRAFT STANDARDS, ARE PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE, AND THE PARTICIPANTS, REPRESENTATIVES, THE CDISC PRESIDENT, THE CDISC BOARD OF DIRECTORS, AND CDISC EXPRESSLY DISCLAIM ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR OR INTENDED PURPOSE, OR ANY OTHER WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, FINAL STANDARDS OR DRAFT STANDARDS, OR CONTRIBUTION.

Limitation of Liability
IN NO EVENT WILL CDISC OR ANY OF ITS CONSTITUENT PARTS (INCLUDING, BUT NOT LIMITED TO, THE CDISC BOARD OF DIRECTORS, THE CDISC PRESIDENT, CDISC STAFF, AND CDISC MEMBERS) BE LIABLE TO ANY OTHER PERSON OR ENTITY FOR ANY LOSS OF PROFITS, LOSS OF USE, DIRECT, INDIRECT, INCIDENTAL, CONSEQUENTIAL, OR SPECIAL DAMAGES, WHETHER UNDER CONTRACT, TORT, WARRANTY, OR OTHERWISE, ARISING IN ANY WAY OUT OF THIS POLICY OR ANY RELATED AGREEMENT, WHETHER OR NOT SUCH PARTY HAD ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

Note: The CDISC Intellectual Property Policy can be found at
http://www.cdisc.org/about/bylaws_pdfs/CDISCIPPolicy-FINAL.pdf