

Accessing and using the metadata from the define.xml

Lex Jansen

Octagon Research Solutions, Inc.

Leading the Electronic Transformation of Clinical R&D

**PharmaSUG 2010,
Orlando, FL**

- What is XML
- Reading XML with SAS (**exercises**)
- What is the define.xml
- Why convert the define .xml to SAS ???
 - printing issue
 - Validating the define.xml
 - Other use cases
- define.xml as a relational data model
- SAS XML mapper
- Converting define.xml to SAS datasets (**exercises**)



What is XML



- **XML** stands for e**X**tensible **M**arkup **L**anguage
- **XML** is a markup language much like HTML, with tags, elements and attributes
- **XML** was designed to transport and store data, not to display data (display and content separated)
- **XML** does not DO anything
- **XML** tags are not predefined. You must define your own (self-descriptive!) tags



- Like **HTML**, **XML** makes use of *tags* (words bracketed by '<' and '>') and *attributes* (of the form name="value")
- **BUT ...**, **HTML** specifies what each tag and attribute means, and often how the text between them will look in a browser
- **XML** uses the tags only to delimit pieces of data, and leaves the interpretation of the data completely to the application that reads it
- In other words, if you see "<p>" in an XML file, do not assume it is a paragraph. Depending on the context, it may be a price, a parameter, a person, (or maybe something that does not start with with a "p"?)



- An XML file is **well-formed** if it conforms to the rules of XML syntax
 - A single element (root element) contains all other elements in the document (define.xml : <ODM>)
 - Elements have to be properly opened and closed
 - Elements do not overlap, e.g. properly nested
 - Attributes are properly quoted
 - The document does not contain illegal characters.
Example: if the left opening bracket “<” is part of the content it should be substituted as “<”
 - A conforming XML parser is not allowed to process an XML document that is not well-formed



- Predefined entities:

Character	Entity
&	&
<	<
>	>
"	"
'	'

Entities **&** and **<** MUST be used within text value of an element or attribute



- An XML file is **valid** if it conforms to a specific **XML schema**
- An **XML schema** is a description of a type of XML document
- A Schema defines constraints on the structure and contents of documents of that type
- A Schema defines allowed elements and attributes, order of elements, overall structure, etc ...
- A schema might describe that the content of a certain element, that contains a datetime value, is only valid if the value conforms to the ISO8601 standard



- SAS can read generic XML files, that have the following characteristics:
- The enclosing root element is comparable to a SAS library
- A second-level element is translated to a data set name
- Other elements within that second level become SAS variables



Reading XML with SAS



The example XML file we use contains data from "**THE HEART OF ROCK & SOUL**, the 1001 Greatest Singles Ever Made" by *Dave Marsh*.

See also:

<http://www.lexjansen.com/marsh>



```
<?xml version="1.0" encoding="ISO-8859-1" ?>
```

root element

```
<heartofrockandsoul>
```

data set

```
<entry>
```

```
<rank>556</rank>
```

```
<artist>The Sheppards</artist>
```

```
<title>Tragic</title>
```

```
<producer>Bunky Sheppard</producer>
```

```
<writer>Kermit Chandler & O.C. Perkins</writer>
```

```
<label>Apex 7762</label>
```

```
<year>1961</year>
```

```
<billboard>Did not make pop charts</billboard>
```

```
</entry>
```

```
<entry>
```

```
<rank>938</rank>
```

```
<artist>Gino Washington</artist>
```

```
<title>Gino is a Coward</title>
```

```
<producer>Sonny Saunders</producer>
```

```
<writer>Ronald Davis</writer>
```

```
<label>Ric Tic</label>
```

```
<year>1963</year>
```

```
<billboard>Did not make pop charts</billboard>
```

```
</entry>
```

```
</heartofrockandsoul>
```

variables

A "rectangular"
XML file

```
* Exercise01-A_ReadXML.sas ;
```

```
FILENAME rocksoul "&WorkShop\xml\heartofrocknsoul-1.xml";
```

```
LIBNAME rs xml XMLFILEREFS=rocksoul;
```

```
PROC CONTENTS DATA=rs._ALL_ VARNUM;  
RUN;
```

```
DATA work.RocknSoul1;  
  SET rs.entry;  
RUN;
```

```
PROC CONTENTS DATA=work.RocknSoul1 VARNUM;  
RUN;
```

```
PROC PRINT DATA=work.RocknSoul1;  
RUN;
```



The CONTENTS Procedure

Data Set Name	RS.ENTRY	Observations	.
Member Type	DATA	Variables	8
Engine	XML	Indexes	0
Created	.	Observation Length	0
Last Modified	.	Deleted Observations	0
Protection		Compressed	NO
Data Set Type		Sorted	NO
Label			
Data Representation	Default		
Encoding	Default		

Variables in Creation Order

#	Variable	Type	Len	Format	Informat	Label
1	BILLBOARD	Char	23	\$23.	\$23.	BILLBOARD
2	YEAR	Num	8	F8.	F8.	YEAR
3	LABEL	Char	14	\$14.	\$14.	LABEL
4	WRITER	Char	44	\$44.	\$44.	WRITER
5	PRODUCER	Char	29	\$29.	\$29.	PRODUCER
6	TITLE	Char	36	\$36.	\$36.	TITLE
7	ARTIST	Char	28	\$28.	\$28.	ARTIST
8	RANK	Num	8	F8.	F8.	RANK

& transformed to &

```
<entry>  
  <rank>556</rank>  
  <artist>The Sheppards</artist>  
  <title>Tragic</title>  
  <producer>Bunky Sheppard</producer>  
  <writer>Kermit Chandler & O.C. Perkins</writer>  
  <label>Apex 7762</label>  
  <year>1961</year>  
  <billboard>Did not make pop charts</billboard>  
</entry>
```

WRITER

```
Norman Whitfield & Barrett Strong  
Chuck Berry  
James Brown  
Brian Holland, Lamont Dozier & Edc  
Isaac Hayes & David Porter  
Nolan Strong and The Diablos  
Nathaniel Mayer & Devora Brown  
Dan Penn & Spooner Oldham  
Arlester (Dyke) Christian  
Chips Moman & Dan Penn  
Fred Parris  
Isaac Hayes & David Porter  
Kermit Chandler & O.C. Perkins  
Dickey Lee  
Ronald Davis
```



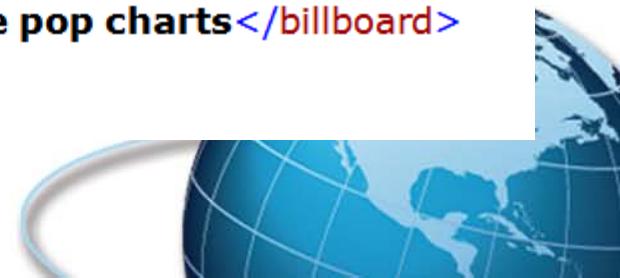
data set



variables



```
<?xml version="1.0" encoding="ISO-8859-1" ?>  
<heartofrockandsoul>  
  <entry>  
    <rank>556</rank>  
    <artist>The Sheppards</artist>  
    <title>Tragic</title>  
    <producer>Bunky Sheppard</producer>  
    <writer>Kermit Chandler & O.C. Perkins</writer>  
    <label>Apex 7762</label>  
    <year>1961</year>  
    <billboard>Did not make pop charts</billboard>  
  </entry>  
  <entry>  
    <rank>938</rank>  
    <artist>Gino Washington</artist>  
    <title>Gino is a Coward</title>  
    <producer>Sonny Saunders</producer>  
    <writer>Ronald Davis</writer>  
    <label>Ric Tic</label>  
    <year>1963</year>  
    <billboard>Did not make pop charts</billboard>  
  </entry>  
</heartofrockandsoul>
```



```
* Exercise01-B_ReadXML.sas ;
```

```
FILENAME rocksoul "&WorkShop\xml\heartofrocknsoul-2.xml";
```

```
LIBNAME rs xml XMLFILEREFF=rocksoul;
```

```
DATA work.RocknSoul2;
```

```
  SET rs.entry;
```

```
RUN;
```

```
PROC PRINT DATA=work.RocknSoul2;
```

```
RUN;
```



```
29 * Excercise01-B_ReadXML.sas ;
30
31 FILENAME rock soul "&WorkShop\xml\heartofrocknsoul-2.xml";
32 LIBNAME rs xml XMLFILEREf=rock soul;
NOTE: Libref RS was successfully assigned as follows:
      Engine:          XML
      Physical Name:
33
34 DATA work.RocknSoul2;
35 SET rs.entry;
ERROR: There is an illegal character in the entity name.
       encountered during XMLMap parsing
       occurred at or near line 8, column 31
ERROR: XML describe error: Internal processing error.
36 RUN;
```

NOTE: The SAS System stopped processing this step because of error
WARNING: The data set WORK.ROCKNSOUL2 may be incomplete. When there were 0 observations and 0 variables.
NOTE: DATA statement used (Total process time):
 real time 0.03 seconds
 cpu time 0.01 seconds



- Characters like the ampersand (&) and the left angle bracket (<) must be escaped: **&**; and **<**;
- To import an XML document that contains non-escaped characters, you can specify the **XMLPROCESS=RELAX** option on the LIBNAME.
- **Note: This is not recommended.**
If an XML document consists of non-escaped characters, the content is **not well-formed**.
- A conforming XML parser is not allowed to process an XML document that is not well-formed



```
* Exercise01-C_ReadXML.sas ;
```

```
FILENAME rocksoul "&WorkShop\xml\heartofrocknsoul-2.xml";
```

```
LIBNAME rs xml XMLFILEREFF=rocksoul XMLPROCESS=RELAX;
```

```
DATA work.RocknSoul2;
```

```
  SET rs.entry;
```

```
RUN;
```

```
PROC PRINT DATA=work.RocknSoul2;
```

```
RUN;
```



```
40 * Excercise01-C_ReadXML.sas ;
41
42 FILENAME rocksoul "&WorkShop\xml\heartofrocknsoul-2.xml";
43 LIBNAME rs xml XMLFILEREf=rocksoul XMLPROCESS=RELAX;
NOTE: Libref RS was successfully assigned as follows:
      Engine:          XML
      Physical Name:
44
45 DATA work.RocknSoul2;
46   SET rs.entry;
47   RUN;
```

```
NOTE: There were 16 observations read from the data set RS.ENTRY.
NOTE: The data set WORK.ROCKNSOUL2 has 16 observations and 8 variables.
NOTE: DATA statement used (Total process time):
      real time          0.09 seconds
      cpu time           0.07 seconds
```

```
48
49 PROC PRINT DATA=work.RocknSoul2;
50   RUN;
```

```
NOTE: There were 16 observations read from the data set WORK.ROCKNSOUL2.
NOTE: PROCEDURE PRINT used (Total process time):
      real time          0.00 seconds
      cpu time           0.00 seconds
```

Let's make rank
an attribute

```
<?xml version="1.0" encoding="ISO-8859-1" ?>  
<heartofrockandsoul>  
  <entry>  
    <rank>556</rank>  
    <artist>The Sheppards</artist>  
    <title>Tragic</title>  
    <producer>Bunky Sheppard</producer>  
    <writer>Kermit Chandler & O.C. Perkins</writer>  
    <label>Apex 7762</label>  
    <year>1961</year>  
    <billboard>Did not make pop charts</billboard>  
  </entry>
```

```
<?xml version="1.0" encoding="ISO-8859-1" ?>  
<heartofrockandsoul>  
  <entry rank="556">  
    <artist>The Sheppards</artist>  
    <title>Tragic</title>  
    <producer>Bunky Sheppard</producer>  
    <writer>Kermit Chandler & O.C. Perkins</writer>  
    <label>Apex 7762</label>  
    <year>1961</year>  
    <billboard>Did not make pop charts</billboard>  
  </entry>
```



```
* Exercise01-D_ReadXML.sas ;
```

```
FILENAME rocksoul "&WorkShop\xml\heartofrocknsoul-3.xml";
```

```
LIBNAME rs xml XMLFILEREF=rocksoul;
```

```
PROC CONTENTS DATA=rs._ALL_ VARNUM;  
RUN;
```

```
DATA work.RocknSoul3;  
  SET rs.entry;  
RUN;
```

```
PROC CONTENTS DATA=work.RocknSoul3 VARNUM;  
RUN;
```

```
PROC PRINT DATA=work.RocknSoul3;  
RUN;
```



The CONTENTS Procedure

```

Data Set Name      RS.ENTRY      Observations      .
Member Type       DATA        Variables         7
Engine            XML          Indexes           0
Created           .            Observation Length 0
Last Modified     .            Deleted Observations 0
Protection        .            Compressed        NO
Data Set Type     .            Sorted            NO
Label
Data Representation Default
Encoding          Default
  
```

Variables in Creation Order

#	Variable	Type	Len	Format	Informat	Label
1	BILLBOARD	Char	23	\$23.	\$23.	BILLBOARD
2	YEAR	Num	8	F8.	F8.	YEAR
3	LABEL	Char	14	\$14.	\$14.	LABEL
4	WRITER	Char	44	\$44.	\$44.	WRITER
5	PRODUCER	Char	29	\$29.	\$29.	PRODUCER
6	TITLE	Char	36	\$36.	\$36.	TITLE
7	ARTIST	Char	28	\$28.	\$28.	ARTIST

RANK ???



What is define.xml




Regulatory landscape



- July 2004 – FDA adds Study Data Specifications v1.0 to draft eCTD Guidance. This specification references the CDISC SDTM for data tabulation datasets

Electronic Common Technical Document (eCTD)

- Draft Guidance for Industry on Providing Regulatory Submissions in Electronic Format--Human Pharmaceutical Applications and Related Submissions. (Posted 8/28/2003)
 - *Federal Register* Notice [\[TXT\]](#) [\[PDF\]](#)
 - [The Draft Guidance](#) 

Specifications

- [eCTD Backbone Files Specification for Module 1](#) 
- [eCTD Backbone File Specification for Modules 2 through 5](#) 
- [eCTD Backbone File Specification for Study Tagging Files](#) 
- [FDA eCTD Table of Contents Headings and Hierarchy](#) 
- [Study Data Specifications](#)  **New!!** (Posted 7/21/2004)



- March 2005 – Study Data Specifications v1.1:
Updates Specifications for Data Set Documentation
 - data definitions
 - annotated case report forms (CRFs)
- *“The specification for the data definitions for datasets provided using the CDISC SDTM is included in the Case Report Tabulation Data Definition Specification (**define.xml**) developed by the CDISC define.xml Team”*
- Data Definition for other data sets follows:
*Providing Regulatory Submissions in Electronic Format – NDA (1999), which is the **define.pdf***



- 2006 – CDISC SDTM / ADaM Pilot Project:
Collaborative Pilot project with FDA and industry to test how well the submission of CDISC compliant data sets and associated metadata meets the needs of both medical and statistical FDA reviewers
- Generation of ICH E3/eCTD clinical study report (CSR) using the CDISC data models
- Data Definition Tables were provided in XML format (CRT-DDS, **define.xml**)



Data Definition Tables in PDF



- 1999 Guidance: sponsor has to document submitted data by including data definition tables ([define.pdf](#)) and annotated case report forms ([blankcrf.pdf](#))

Datasets For Study 1234		
Dataset	Description of dataset	Location
DEMO	Demographics	crt/datasets/1234/demo.xpt
INCLUDE	Inclusion criteria	crt/datasets/1234/include.xpt



- 1999 Guidance: sponsor has to document submitted data by including data definition tables ([define.pdf](#)) and annotated case report forms ([blankcrf.pdf](#))

Study 1234 - Demographics				
Variable	Label	Type	Codes	Comments
PATID	Patient identification	char		
STUDY	Study number	char		
CENTER	Study center	char		
TRT	Assigned treatment group	num	0= placebo 5= 5mg/day	
SEX	Sex of subject	char	f = female m = male	
BDATE	Birth date	date		Demographics page 1
WEIGHT	Weight in kg	num		



Data Definition Tables in XML



- As of January 1, 2008: follow the eCTD guidance and document submitted data by including data definition tables (**define.xml**) and annotated case report forms (blankcrf.pdf)

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<?xml-stylesheet type="text/xsl" href="define1-0-0.xsl"?>
<ODM xmlns="http://www.cdisc.org/ns/odm/v1.2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:def="http://www.cdisc.org/ns/def/v1.0"
xsi:schemaLocation="http://www.cdisc.org/ns/odm/v1.2 define1-0-0.xsd"
FileOID="Studycdisc01"
ODMVersion="1.2" FileType="Snapshot"
CreationDateTime="2007-04-09T12:24:09">
- <Study OID="cdisc01">
- <GlobalVariables>
  <StudyName>CDISC01</StudyName>
  <StudyDescription>CDISC01 Test Study.</StudyDescription>
  <ProtocolName>CDISC01</ProtocolName>
</GlobalVariables>
<MetaDataVersion OID="CDISC.SDTM.3.1.1"
Name="Study CDISC01, Data Definitions"
Description="Study CDISC01, Data Definitions"
def:DefineVersion="1.0.0"
def:StandardName="CDISC SDTM"
def:StandardVersion="3.1.1">
...
<ItemGroupDef OID="MH"
Name="MH" Repeating="Yes" IsReferenceData="No"
Purpose="Tabulation" def:Label="Medical History"
def:Structure="One record per medical history event per subject"
def:DomainKeys="STUDYID, USUBJID, MHCAT, MHTERM, MHSTDTC"
def:Class="EVENTS"
def:ArchiveLocationID="Location.MH">
  <ItemRef ItemOID="STUDYID" OrderNumber="1" Mandatory="Yes"
Role="IDENTIFIER" RoleCodeListOID="RoleCodeList" />
...
  <def:leaf ID="Location.MH" xlink:href="mh.xpt">
    <def:title>mh.xpt</def:title>
  </def:leaf>
</ItemGroupDef>
```



- As of January 1, 2008: follow the eCTD guidance and document submitted data by including data definition tables (**define.xml**) and annotated case report forms (blankcrf.pdf)

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<?xml-stylesheet type="text/xml" href="define1-0-0.xml"?>
<ODH xmlns="http://www.cdisc.org/ns/odm/v1.2"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:def="http://www.cdisc.org/ns/def/v1.0"
  xsi:schemaLocation="http://www.cdisc.org/ns/odm/v1.2 define1-0-0.xsd"
  FileOID="StudyCDISC01"
  ODMVersion="1.2" FileType="Snapshot"
  CreationDateTime="2007-04-09T12:24:09">
  - <Study OID="cdisc01">
    - <GlobalVariables>
      <StudyName>CDISC01</StudyName>
      <StudyDescription>CDISC01 Test Study.</StudyDescription>
      <ProtocolName>CDISC01</ProtocolName>
    </GlobalVariables>
    <MetaDataVersion OID="CDISC.SDTH.3.1.1"
      Name="Study CDISC01, Data Definitions"
      Description="Study CDISC01, Data Definitions">
      <ItemGroupDef OID="MH"
        Name="MH" Repeating="Yes" IsReferenceData="No"
        Purpose="Tabulation" def:Label="Medical History"
        def:Structure="One record per medical history event per subject"
        def:DomainKeys="STUDYID, USUBJID, MHCAT, MHTERM, MHSTDTC"
        def:Class="EVENTS"
        def:ArchiveLocationID="Location.MH">
        <ItemRef ItemOID="STUDYID" OrderNumber="1" Mandatory="Yes"
          Role="IDENTIFIER" RoleCodeListOID="RoleCodeList" />
        ...
        <def:leaf ID="Location.MH" xlink:href="mh.xpt">
          <def:title>mh.xpt</def:title>
        </def:leaf>
      </ItemGroupDef>
    </MetaDataVersion>
  </Study>
</ODH>
```



- Case Report Tabulation Data Specification (CRT-DDS, define.xml)
- Production version: **1.0.0**
- Based on version ODM version **1.2.1**
- **CRT-DDS version 1.0.0** is currently the only production version
- Maintained by CDISC's **XML Technologies Team** (formerly known as the ODM team)
- New draft version of define.xml expected in 2010 with additional metadata support



<http://www.cdisc.org/define-xml>



ABOUT CDISC | **STANDARDS** | RESOURCES | NEWS | EDUCATION & EVENTS

STANDARDS

SDTM

Operational Data Model

Define.XML

Study/Trial Design Model

LAB

ADaM

Protocol

Terminology

CDASH

SEND

CDISC SHARE

Therapeutic Area Standards

Define.XML

FDA Adds CDISC ODM Define.xml to Study Data Specifications

The FDA has now included the CDISC Case Report Tabulation Data Definition Specification (define.xml), which is based on the CDISC ODM, as part of the eCTD Study Data Specifications for the eCTD for submissions using the SDTM. The revised specifications are [available here](#).

Case Report Tabulation Data Definition Specification (CRT-DDS, also called define.xml) Final Version 1.0

CRT-DDS Released for Implementation February 10, 2005.

The CDISC define.xml Team has published the Case Report Tabulation Data Definition Specification (define.xml) Version 1.0 for

Case Report Tabulation Data Definition Specification (define.xml) Prepared by the CDISC define.xml Team

Principal Editor: William Qubeck
Principal Contributors: Sally Cassells, Anthony Friebel, and the define.xml team

Notes to Readers

This version of the Case Report Tabulation Data Definition Specification supersedes all prior versions. Version 1.0.0 reflects changes from a comment period through the Health Level 7 (HL7) Regulated Clinical Research Information Management Technical Committee (RCRIM) in December 2003 (www.hl7.org) and CDISC's website in September 2004 as well as the work done by the define.xml team to add functionality, features, and additional documentation.

Version 1.0.0 incorporated the applicable comments, suggestions, and corrections received from the two comment periods specified above and is the official implementation version.

Revision History

Date	Version	Summary of Changes	Primary Author
2005-02-05	1.0.0	This is the official implementation version of the Case Report Tabulation Data Definition specification.	The define.xml team
2005-02-09	1.0.0	Administrative update.	Anthony Friebel, William Qubeck, Sally Cassells



Clinical Data Interchange Standards Consortium

Specification for the Operational Data Model (ODM)

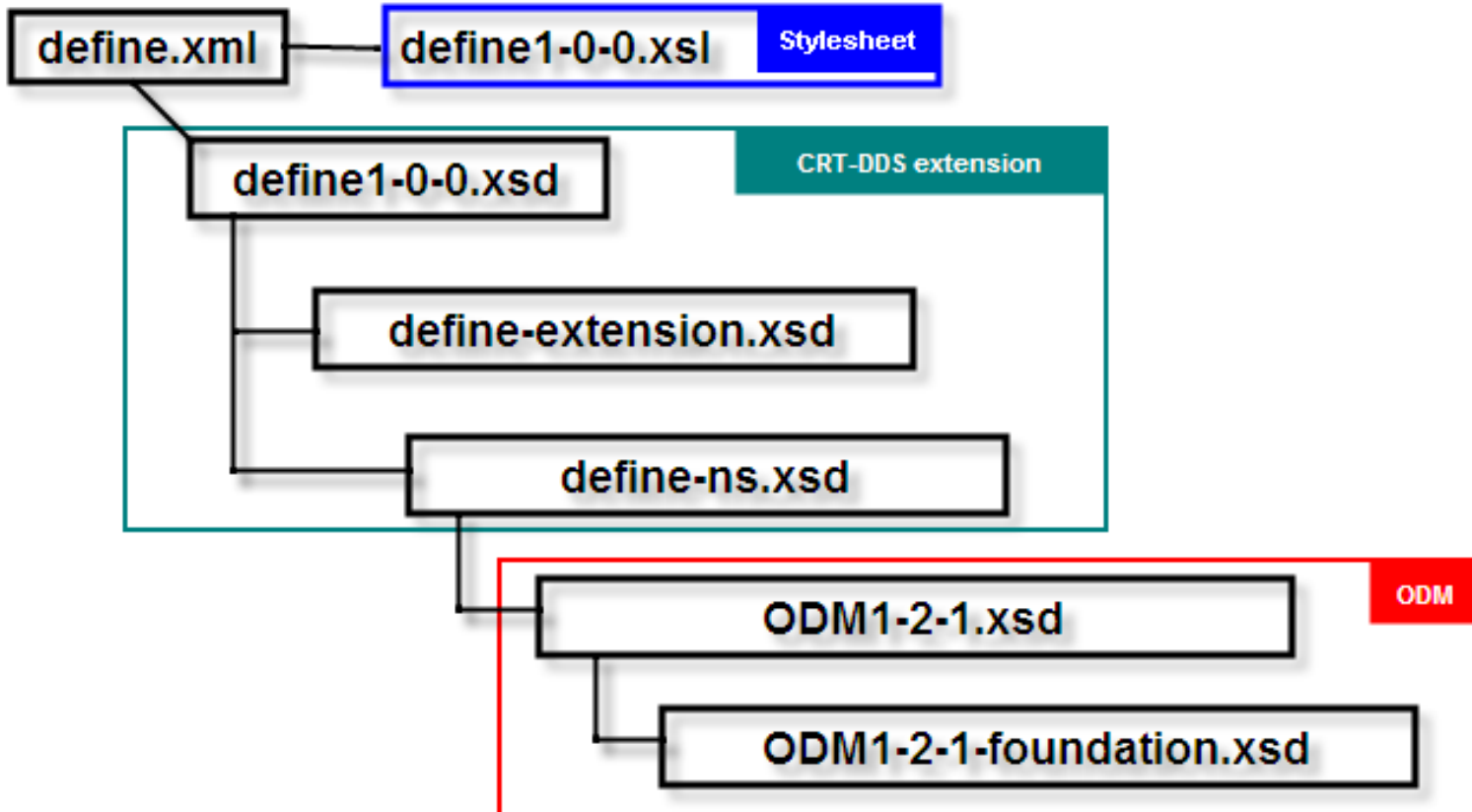
Version 1.2
Source File: ODM-1.2.5.adtd
Last Update: 19 Dec 2003 10:10 AM

Copyright © CDISC 2003 This document is the property of CDISC Inc. This document can be freely used and reproduced without limitation as long as (1) it is not modified, and (2) the entire copyright statement is included in the copy. Modifications to this document can only be made with written consent of CDISC Inc.

An official copy of this document is available at <http://www.cdisc.org/models/odm/v1.2/ODM1-2-0.html>.

Table of Contents

- [1 Introduction \(non-normative\)](#)
- [2 General Issues](#)
 - [2.1 Content of the Standard](#)
 - [2.2 File Conformity](#)
 - [2.3 System Conformity](#)
 - [2.4 Vendor Extensions](#)
 - [2.5 Changes from Previous Versions \(non-normative\)](#)
 - [2.6 Entities and Elements](#)
 - [2.7 Clinical Data Keys](#)
 - [2.8 Single Files and Collections](#)
 - [2.9 Transactions](#)
 - [2.10 Element Ordering](#)
 - [2.11 Element Identifiers and References](#)
 - [2.12 Syntax Notation](#)
 - [2.13 Data Formats](#)
- [3 General Elements](#)
 - [3.1 ODM](#)
 - [3.2 Study](#)
 - [3.3 Global Variables](#)
 - [3.4 StudyName](#)
 - [3.5 StudyDescription](#)
 - [3.6 ProtocolName](#)
 - [3.7 Basic Definitions](#)
 - [3.8 MeasurementUnit](#)
 - [3.9 Symbol](#)
 - [3.10 TranslatedText](#)
- [4 Metadata Elements](#)

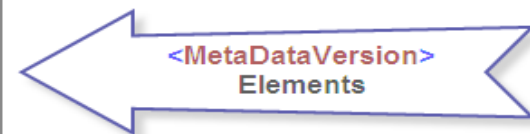


1 <?xml version="1.0" encoding="ISO-8859-1" ?>
<?xml-stylesheet type="text/xsl" href="define1-0-0.xsl"?>

2 <ODM xmlns="http://www.cdisc.org/ns/odm/v1.2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:def="http://www.cdisc.org/ns/def/v1.0"
xsi:schemaLocation="http://www.cdisc.org/ns/odm/v1.2 define1-0-0.xsd"
FileOID="Studydisc01"
ODMVersion="1.2" FileType="Snapshot"
CreationDateTime="2007-04-09T12:24:09">
3 <Study OID="cdisc01">
4 <GlobalVariables>
<StudyName>CDISC01</StudyName>
<StudyDescription>CDISC01 Test Study.</StudyDescription>
<ProtocolName>CDISC01</ProtocolName>
</GlobalVariables>
5 <MetaDataVersion OID="CDISC.SDTM.3.1.1"
Name="Study CDISC01, Data Definitions"
Description="Study CDISC01, Data Definitions"
def:DefineVersion="1.0.0"
def:StandardName="CDISC SDTM"
def:StandardVersion="3.1.1">
6

< def:AnnotatedCRF >
< def:SupplementalDoc >
< def:leaf >
< def:ComputationMethod >
< def:ValueListDef >
< ItemGroupDef >
< ItemDef >
< CodeList >

</MetaDataVersion>
</Study>
</ODM>



Displaying the define.xml



- From the CDISC SDS Metadata Team (2007):
define.xml + XSL style sheet = html

Study CDISC01, Data Definitions

Home | Print | Page | Tools

- Annotated Case Report Form
- Datasets
- Value Level Metadata
- Computational Algorithms
- Controlled Terminology

Datasets for Study CDISC01						
Dataset	Description	Class	Structure	Purpose	Keys	Location
SV	Subject Visits	TRIAL DESIGN	One record per subject per actual visit	Tabulation	STUDYID, USUBJID, SVSTDTC, SVENDTC, VISITNUM, SVUPDES	sv.xpt
TA	Trial Arms	TRIAL DESIGN	One record per element per arm	Tabulation	STUDYID, ARMCD, TAETORD	ta.xpt
TE	Trial Elements	TRIAL DESIGN	One record per element	Tabulation	STUDYID, ETCD	te.xpt
TI	Trial Inclusion	TRIAL DESIGN	One record per I/E criterion	Tabulation	STUDYID, IETESTCD	ti.xpt
TS	Trial Summary	TRIAL DESIGN	One record per parameter value	Tabulation	STUDYID, TSPARMCD, TSVAL	ts.xpt
TV	Trial Visits	TRIAL DESIGN	One record per planned visit per arm	Tabulation	STUDYID, VISITNUM, ARMCD	tv.xpt
CO	Comments	SPECIAL PURPOSE	One record per comment per subject	Tabulation	STUDYID, USUBJID, COSEQ, IDVAR, IDVARVAL	co.xpt
DM	Demographics	SPECIAL PURPOSE	One record per subject	Tabulation	STUDYID, USUBJID	dm.xpt
CM	Concomitant Medications	INTERVENTIONS	One record per medication intervention episode	Tabulation	STUDYID, USUBJID, CMCAT, CMTRT, CMSTDTC, CMENDTC,	cm.xpt

- From the CDISC SDS Metadata Team (2007):
define.xml + XSL style sheet = html

Study CDISC01, Data Definitions

Home | Print | Page | Tools

Annotated Case Report Form

Datasets

- Subject Visits (SV)
- Trial Arms (TA)
- Trial Elements (TE)
- Trial Inclusion (TI)
- Trial Summary (TS)
- Trial Visits (TV)
- Comments (CO)
- Demographics (DM)
- Concomitant Medications (CM)
- Exposure (EX)
- Adverse Events (AE)
- Disposition (DS)
- Medical History (MH)
- Drug Accountability (DA)
- ECG (EG)
- Inclusion/Exclusion Exceptions (IE)
- Laboratory Tests (LB)
- Physical Exam (PE)
- Questionnaires (QS)
- Subject Characteristics (SC)
- Vital Signs (VS)
- Supplemental Qualifier (SUPPAE)
- Supplemental Qualifier (SUPPCM)
- Supplemental Qualifier

Concomitant Medications Dataset (CM)						cm.xpt
Variable	Label	Type	Controlled Terminology	Origin	Role	Comment
STUDYID	Study Identifier	text		CRF Page 3	IDENTIFIER	
DOMAIN	Domain Abbreviation	text		DERIVED	IDENTIFIER	
USUBJID	Unique Subject Identifier	text		SPONSOR DEFINED	IDENTIFIER	Concatenation of STUDYID.SUBJID
CMSEQ	Sequence Number	integer		DERIVED	IDENTIFIER	Sequential number uniquely identifying the records within the domain by USUBJID and generated using the key sequence from the domain level metadata
CMSPID	Sponsor-Defined Identifier	text		SPONSOR DEFINED	IDENTIFIER	ID of original SAS dataset
CMTRT	Reported Name of Drug, Med, or Therapy	text		CRF Pages 6 , 35	TOPIC	
CMMODIFY	Modified Reported Name	text		SPONSOR DEFINED	SYNONYM QUALIFIER	
CMDECOD	Standardized Medication Name	text	DRUGDICT F	DERIVED	SYNONYM QUALIFIER	WHODRUG Version 2002/04
CMCAT	Category for Medication	text	CMCAT F	CRF Pages 6 , 35	GROUPING QUALIFIER	Sponsor controlled terminology

Why convert the define .xml to SAS ???



Define.xml in the SDTM/ADaM CDISC FDA Pilot – Printing Issue



- CDISC SDTM / ADaM Pilot project report:

*“A major **issue** identified by the regulatory review team was the **difficulty in printing the Define file**. The style sheet used in the pilot submission package was developed with the **primary target of web browser rendering**, which is not readily suited to printing. Reviewers who attempted to print the Define file found that the file did not fit on portrait pages, that page breaks were not clean, and that printing only a portion of the file was difficult. Opening the document in another application (e.g., Microsoft Word) provided a work-around, but was not an option that was user friendly or efficient.”*



- CDISC SDTM / ADaM Pilot project report:

*“This problem could be viewed as an **implementation issue** that sponsors will need to handle, after discussing the issue with their FDA reviewers. For example, a sponsor might choose to provide two versions of the style sheet – **XML for viewing** and **PDF for printing**. Ideally, a reminder of the issue would be included somewhere in the CRT-DDS guidance (e.g., a note that consideration be given to how the sponsor will respond to a request from reviewers for a print-friendly version of the style sheet). It should be noted that the **regulatory review team for the pilot project emphasized that the ability to print the document would be essential for the future use of XML files.**”*



Printing the define.xml



- The **PDF** format is the *de facto standard* for **printable** documents on the web
- PDF is platform independent (no browser issues ...)
- How can we create a PDF file from the define.xml ???



SAS Solution:

- Convert the XML hierarchy to a relational data model in the form of (2-dimensional) SAS data sets
- Once we have the define.xml content in SAS datasets, we can use SAS to create a PDF rendition (with ODS PDF) (which is an easy task for a SAS programmer)



Validating the define.xml



Some process used Metadata
(data sets, variables, codelists)
to create:

- define.xml
- SAS transport files



define.xml



Validating the define.xml:

1. well-formedness
2. Against Schema
3. Against CRT-DDS Specification
4. Against SAS transport files
5. Against SDTM spec (“mandatory”)



Validating the define.xml:

1. well-formedness
2. Against Schema
3. Against CRT-DDS Specification
4. Against SAS transport files
5. Against SDTM spec (“mandatory”)

Many XML based tools can do this

XML schema (1.0) can not do this.
Schema 1.1 ?? Schematron ??

XML based tools ?????



Define.xml -> SAS datasets: Validation

Some process used Metadata
(data sets, variables, codelists)
to create:

- define.xml
- SAS transport files



define.xml



Use SAS XML Mapper to convert
define.xml to SAS data sets



define.xml as
SAS datasets

VALIDATION



use SAS ODS
to create define.pdf



Other examples of using the define.xml metadata



- Use codelist information (codes/decodes) to create a PROC FORMAT

```
<CodeList OID="SEX" Name="SEX" DataType="text" SASFormatName="$SEX">  
  <CodeListItem CodedValue="F">  
    <Decode>  
      <TranslatedText xml:lang="en">FEMALE</TranslatedText>  
    </Decode>  
  </CodeListItem>  
  <CodeListItem CodedValue="M">  
    <Decode>  
      <TranslatedText xml:lang="en">MALE</TranslatedText>  
    </Decode>  
  </CodeListItem>  
  <CodeListItem CodedValue="U">  
    <Decode>  
      <TranslatedText xml:lang="en">UNKNOWN</TranslatedText>  
    </Decode>  
  </CodeListItem>  
</CodeList>
```

- Use variable information (type, length, label) to create zero-observation datasets that can serve as data conversion targets

```
<ItemGroupDef OID="DM"  
  Name="DM"  
  Repeating="No"  
  IsReferenceData="No"  
  SASDatasetName="DM"  
  Purpose="Tabulation"  
  def:Label="Demographics"
```

```
<ItemDef OID="DM.SEX"  
  Name="SEX"  
  DataType="text"  
  Length="1"  
  Origin="CRF Page 3"  
  def:Label="Sex"  
>
```

```
<ItemRef ItemOID="DM.SEX"  
  OrderNumber="11"  
  Mandatory="Yes"  
  Role="RESULT QUALIFIER"  
  RoleCodeListOID="RoleCodeList"/>  
<ItemRef ItemOID="DM.RACE"  
  OrderNumber="12"  
  Mandatory="No"  
  Role="RESULT QUALIFIER"  
  RoleCodeListOID="RoleCodeList"/>
```



define.xml as a relational model



- **How to Convert** the XML hierarchy to a relational data model in the form of (2-dimensional) SAS data sets
- **Exercise02_ImportDefine_Naiveve.sas**

```
FILENAME define '&WorkShop\tabulations\define.xml';  
LIBNAME define xml XMLFILEREf=define;  
  
PROC CONTENTS DATA=define._ALL_ VARNUM;  
RUN;
```



```
2
3 FILENAME define "&WorkShop\tabulations\define.xml";
4 LIBNAME define xml XMLFILEREf=define;
NOTE: Libref DEFINE was successfully assigned as follows:
      Engine:      XML
      Physical Name: DEFINE
5
6 PROC CONTENTS DATA=define._ALL_ VARNUM;
7 RUN;
```

ERROR: XML describe error: XML data is not in a format supported natively by the XML libname engine. Files of this type usually require an XMLMap to be input properly: .

NOTE: Statements not processed because of errors noted above.



- **How to Convert** the XML hierarchy to a relational data model in the form of (2-dimensional) SAS data sets
- **Solution: SAS XML Mapper**
- **SAS XML Mapper:**
free stand-alone Java client application available on the SAS product distribution disks
- Uses XPATH to create a MAP file that maps hierarchical XML to rows and columns in SAS



- In SAS:

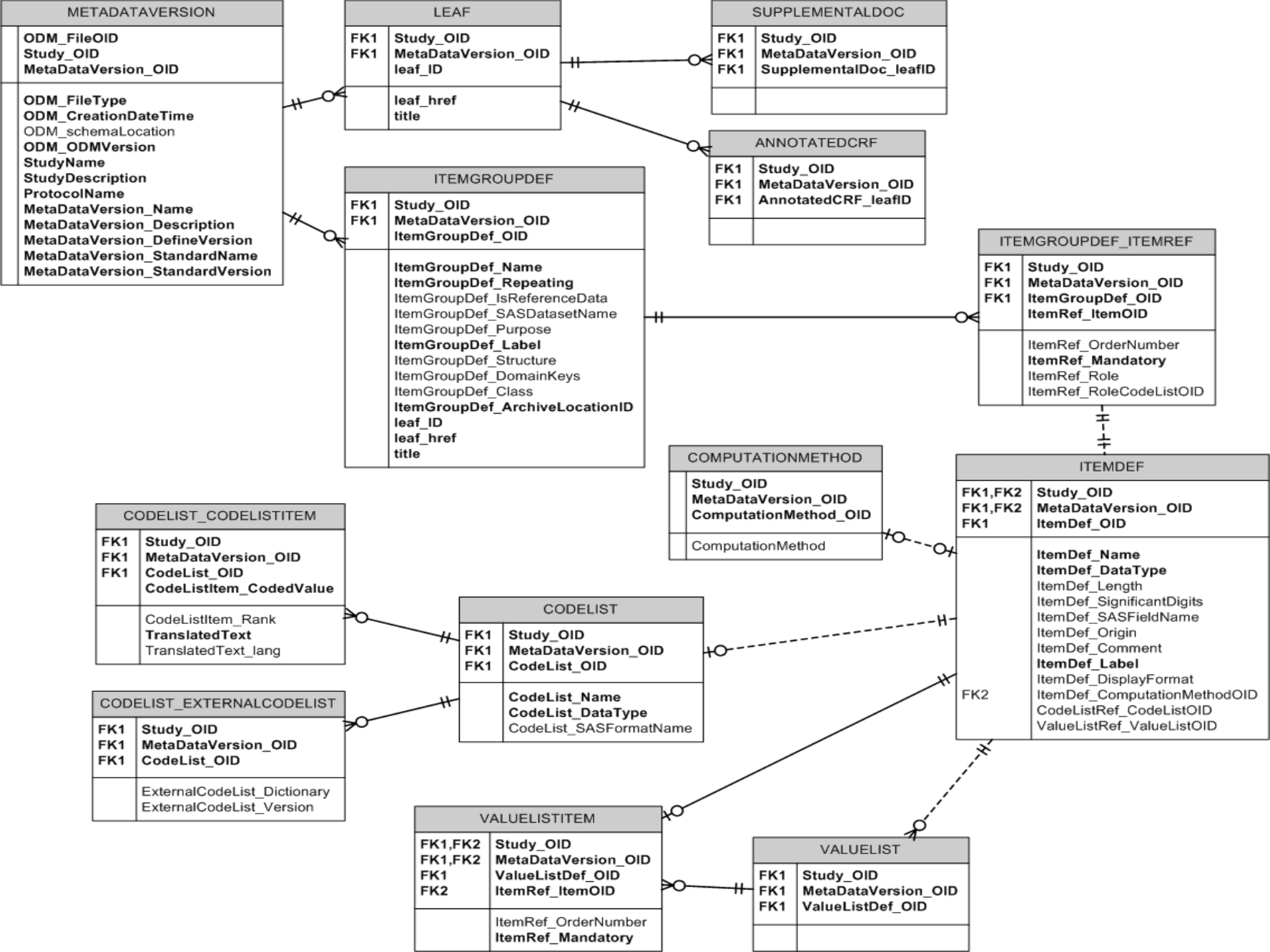
```
FILENAME DEFINE "C:\HOW\jansen\tabulations\define.xml";  
FILENAME SXLEMAP "C:\HOW\jansen\maps\define.map";  
LIBNAME DEFINE XML XMLMAP=SXLEMAP access=READONLY;  
  
PROC COPY IN=define OUT=outlib;  
RUN;
```

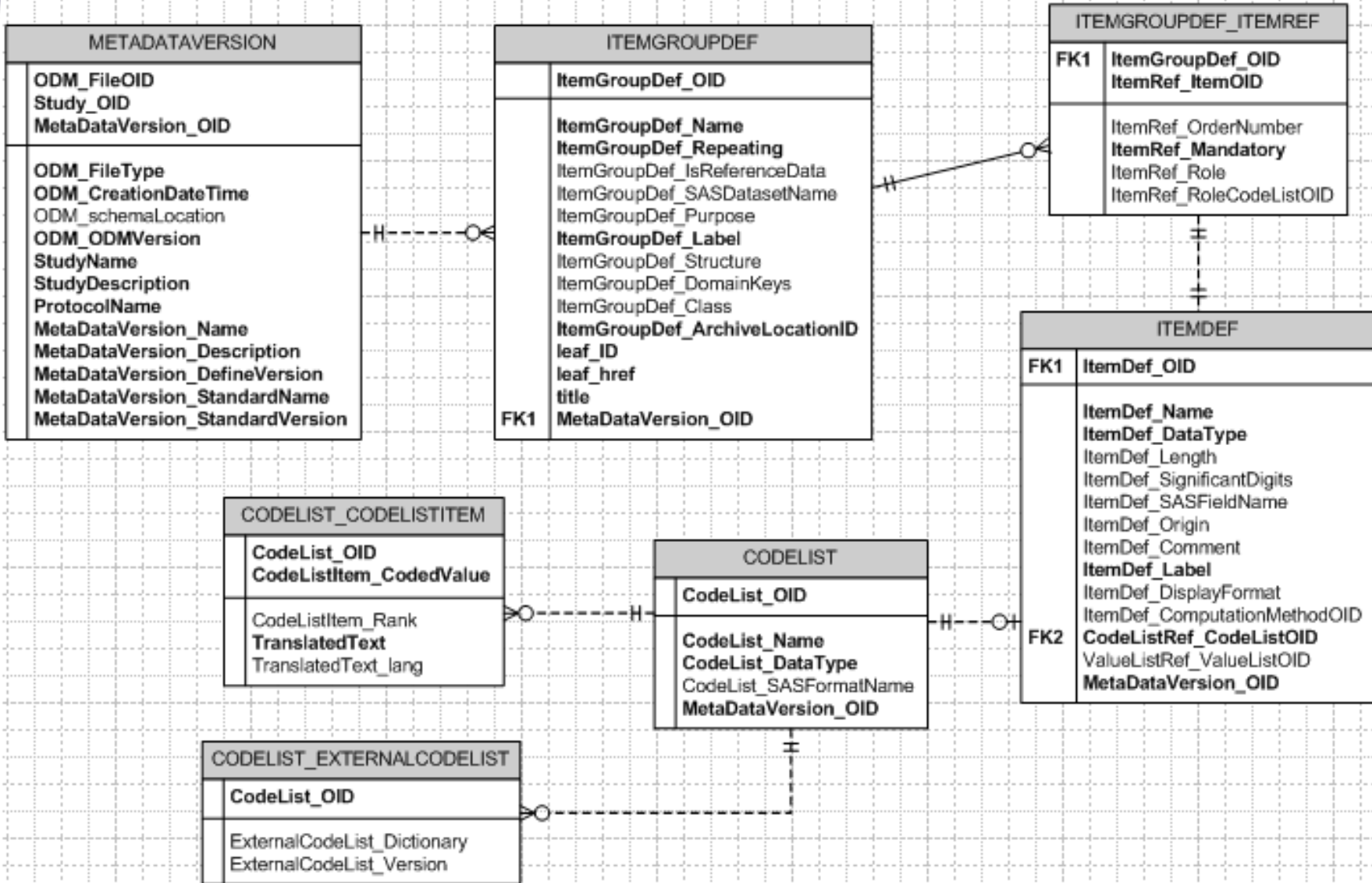


- Solution: SAS XML Mapper + SAS ODS PDF
- You will need to create a relational data model to convert the XML hierarchy to the 2-dimensional SAS data set

```
<ItemGroupDef OID="MH"  
  Name="MH" Repeating="Yes" IsReferenceData="No"  
  Purpose="Tabulation" def:Label="Medical History"  
  def:Structure="One record per medical history event per subject"  
  def:DomainKeys="STUDYID, USUBJID, MHCAT, MHTERM, MHSTDTC"  
  def:Class="EVENTS" def:ArchiveLocationID="Location.AE">  
  <ItemRef ItemOID="STUDYID" OrderNumber="1" Mandatory="Yes"  
    Role="IDENTIFIER" RoleCodeListOID="RoleCodeList" />  
  ...  
  <def:leaf ID="Location.MH" xlink:href="mh.xpt">  
    <def:title>mh.xpt</def:title>  
  </def:leaf>  
</ItemGroupDef>  
...  
<ItemDef OID="STUDYID" Name="STUDYID" DataType="text" Length="7"  
  Origin="CRF Page 3" def:Label="Study Identifier" />
```







SAS XML Mapper





Condensed Full Schema

- MetaDataVersion [1] {1}
 - Attributes [1] {1}
 - def.AnnotatedCRF [1] {1}
 - def.leaf [1] {1}
 - def.ComputationMethod [2] {2}
 - def.ValueListDef [19] {19}
 - ItemGroupDef [28] {28}
 - ItemDef [389] {389}**
 - Attributes [1] {389}
 - OID [1] {389}
 - Name [1] {389}
 - DataType [1] {389}
 - Length [1] {389}
 - Origin [1] {389}
 - Comment [1] {107}
 - def.Label [1] {389}
 - SignificantDigits [1] {14}
 - def.ComputationMethodOID [1] {2}
 - CodeListRef [1] {62}
 - Attributes [1] {62}
 - CodeListOID [1] {62}
 - def.ValueListRef [1] {19}
 - Attributes [1] {19}
 - ValueListOID [1] {19}
 - CodeList [29] {29}

define.xml structure

Properties Format Condition Enumeration Class XMLMap Settings Output

Name: ItemDef_OID
 Description: OID
 Path: /ODM/Study/MetaDataVersion/ItemDef/@OID
 End Path: Begin/End
 Retain Replace

- ItemGroupDef
 - ItemGroupDef_ItemRef
 - ItemDef
 - Study_OID
 - MetaDataVersion_OID
 - ItemDef_OID**
 - ItemDef_Name
 - ItemDef_DataType
 - ItemDef_Length
 - ItemDef_SignificantDigits
 - ItemDef_SASFieldName
 - ItemDef-Origin
 - ItemDef_Comment
 - ItemDef_Label
 - ItemDef_DisplayFormat
 - ItemDef_ComputationMethodOID
 - CodeListRef_CodeListOID
 - ValueListRef_ValueListOID

SAS data set Metadata

Table: ItemDef Row: 24 / 389 Columns: 6 / 15

⚠ SAS formats and informats are not applied to this view.

	ItemDef_OID	ItemDef_Name	ItemDef_D...	ItemDef_Le...	ItemDef_S...	ItemDef...	ItemDef-Origin	ItemDef_Comment
16	SUPPVS.QNAM.VSCSI...	VSCSIND	text	2			DERIVED	Only created if value qualifies as pote
17	AE.AESEQ	AESEQ	integer	1			DERIVED	Sequential number uniquely identifiyir
18	AE.AESPID	AESPID	text	2			SPONSOR DEFINED	ID of original SAS dataset
19	AE.AETERM	AETERM	text	25			CRF Page 34	
20	AE.AEMODIFY	AEMODIFY	text	9			SPONSOR DEFINED	
21	AE.AEDECOD	AEDECOD	text	18			DERIVED	MedDRA version 8.0
22	AE.AEBODSYS	AEBODSYS	text	52			DERIVED	MedDRA version 8.0
23	AE.AESEV	AESEV	text	8			CRF Page 34	

Define.xml -> SAS datasets

itemgroupdef.sas7bdat

	Study_OID	MetaDataVersion_OID	ItemGroup_Def_OID	ItemGroup_Def_Name	ItemGroup_Def_Repea	ItemGroup_upDef_I	ItemGroupDef_Purpose	ItemGroupDef_Label	ItemGroupDef_St	ItemGroupDef_Doma	ItemGroup	ItemGr
11	cdisc01	CDISC.SDTM.3.1.	AE	AE	Yes	No	Tabulation	Adverse Events	One record per	STUDYID, USUBJID,	EVENTS	Locati
12	cdisc01	CDISC.SDTM.3.1.	DS	DS	Yes	No	Tabulation	Disposition	One record per	STUDYID, USUBJID,	EVENTS	Locati
13	cdisc01	CDISC.SDTM.3.1.	MH	MH	Yes	No	Tabulation	Medical History	One record per	STUDYID, USUBJID,	EVENTS	Locati
14	cdisc01	CDISC.SDTM.3.1.	DA	DA	Yes	No	Tabulation	Drug Accountability	One record per	STUDYID, USUBJID,	FINDINGS	Locati
15	cdisc01	CDISC.SDTM.3.1.	EG	EG	Yes	No	Tabulation	ECG	One record per	STUDYID, USUBJID,	FINDINGS	Locati

itemgroupdef_itemref.sas7bdat

	Study_OID	MetaDataVersion_OID	ItemGroup_Def_OID	ItemRef_ItemOID	ItemRef_OrderNumber	ItemRef_Mandatory	ItemRef_Role	ItemRef_RoleCodeListOID
107	cdisc01	CDISC.SDTM.3.1.1	AE	DOMAIN	2	Yes	IDENTIFIER	RoleCodeList
108	cdisc01	CDISC.SDTM.3.1.1	AE	USUBJID	3	Yes	IDENTIFIER	RoleCodeList
109	cdisc01	CDISC.SDTM.3.1.1	AE	AE.AESEQ	4	Yes	IDENTIFIER	RoleCodeList
110	cdisc01	CDISC.SDTM.3.1.1	AE	AE.AESPID	5	No	IDENTIFIER	RoleCodeList
111	cdisc01	CDISC.SDTM.3.1.1	AE	AE.AETERM	6	Yes	TOPIC	RoleCodeList
112	cdisc01	CDISC.SDTM.3.1.1	AE	AE.AEMODIFY	7	No	SYNONYM QUALIFI	RoleCodeList
113	cdisc01	CDISC.SDTM.3.1.1	AE	AE.AEDECOD	8	Yes	SYNONYM QUALIFI	RoleCodeList

itemdef.sas7bdat

	Study_OID	MetaDataVersion_OID	ItemDef_OID	ItemDef_Name	ItemDef_DataType	ItemDef_Length	ItemDef_Origin	ItemDef_Comment	ItemDef_Label	ItemDef_DisplayFormat	ItemDef_ComputationMethodOID	CodeListRef_CodeListOID
18	cdisc01	CDISC.SDTM.3.1.1	AE.AESPID	AESPID	text	2	SPONSOR DEFINED	ID of orig	Sponsor-Defined Ident			
19	cdisc01	CDISC.SDTM.3.1.1	AE.AETERM	AETERM	text	25	CRF Page 34		Reported Term for the			
20	cdisc01	CDISC.SDTM.3.1.1	AE.AEMODIFY	AEMODIFY	text	9	SPONSOR DEFINED		Modified Reported Ter			
21	cdisc01	CDISC.SDTM.3.1.1	AE.AEDECOD	AEDECOD	text	18	DERIVED	MedDRA ver	Dictionary-Derived Te			AEDICT_F
22	cdisc01	CDISC.SDTM.3.1.1	AE.AEBODSYS	AEBODSYS	text	52	DERIVED	MedDRA ver	Body System or Organ			
23	cdisc01	CDISC.SDTM.3.1.1	AE.AESEV	AESEV	text	8	CRF Page 34		Severity/Intensity			AESEVRF
24	cdisc01	CDISC.SDTM.3.1.1	AE.AESER	AESER	text	1	CRF Page 34		Serious Event			NY
25	cdisc01	CDISC.SDTM.3.1.1	AE.AEACN	AEACN	text	30	CRF Page 34		Action Taken with Stu			AEACTF

```
<TABLE name="ItemGroupDef">  
  <TABLE-DESCRIPTION>ItemGroupDef</TABLE-DESCRIPTION>  
  <TABLE-PATH syntax="XPath">/ODM/Study/MetaDataVersion/ItemGroupDef</TABLE-PATH>  
  
  <COLUMN name="MetaDataVersion_OID" retain="YES">  
    <PATH syntax="XPath">/ODM/Study/MetaDataVersion/@OID</PATH>  
    <DESCRIPTION>OID</DESCRIPTION>  
    <TYPE>character</TYPE>  
    <DATATYPE>string</DATATYPE>  
    <LENGTH>100</LENGTH>  
  </COLUMN>  
  
  <COLUMN name="ItemGroupDef_OID">  
    <PATH syntax="XPath">/ODM/Study/MetaDataVersion/ItemGroupDef/@OID</PATH>  
    <DESCRIPTION>OID</DESCRIPTION>  
    <TYPE>character</TYPE>  
    <DATATYPE>string</DATATYPE>  
    <LENGTH>100</LENGTH>  
  </COLUMN>
```

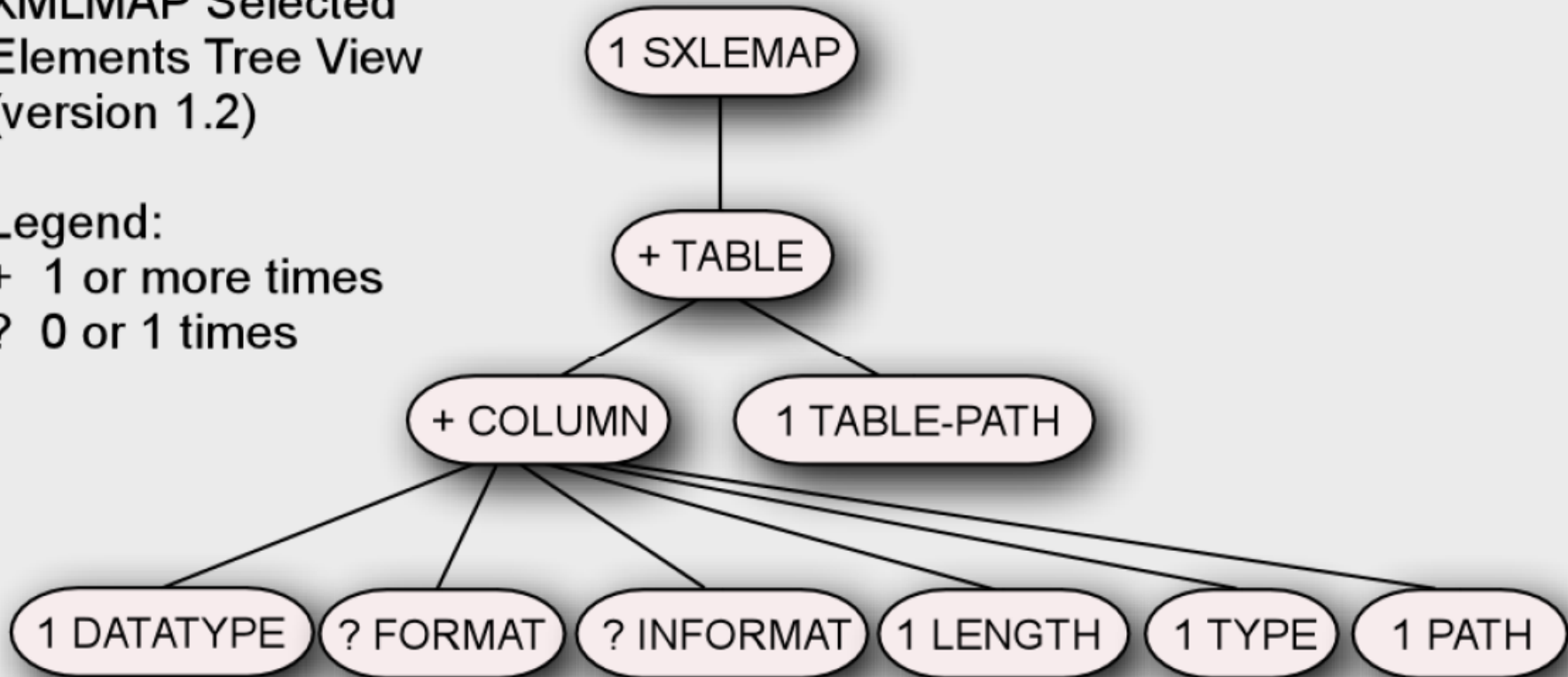


XMLMAP Selected Elements Tree View (version 1.2)

Legend:

+ 1 or more times

? 0 or 1 times



Excercises



PROC FORMAT from the define.xml

```
PROC FORMAT FMTLIB;  
  VALUE $FRAME  
    "S" = "SMALL"  
    "M" = "MEDIUM"  
    "L" = "LARGE"  
    "XL" = "EXTRA LARGE"  
  ;  
  VALUE $SEX  
    "F" = "FEMALE"  
    "M" = "MALE"  
    "U" = "UNKNOWN"  
  ;  
  VALUE SMKCLAS  
    1 = "NEVER SMOKED"  
    2 = "SMOKER"  
    3 = "EX SMOKER"  
  ;  
RUN;
```



DATASET TEMPLATES from the define.xml

```
CREATE TABLE DM(LABEL="Demographics") (  
  STUDYID CHAR(7) LABEL="Study Identifier",  
  DOMAIN CHAR(2) LABEL="Domain Abbreviation",  
  USUBJID CHAR(14) LABEL="Unique Subject Identifier",  
  SUBJID CHAR(6) LABEL="Subject Identifier for the Study",  
  RFSTDTC CHAR(10) LABEL="Subject Reference Start Date/Time",  
  RFENDTC CHAR(10) LABEL="Subject Reference End Date/Time",  
  SITEID CHAR(3) LABEL="Study Site Identifier",  
  BRTHDTC CHAR(10) LABEL="Date/Time of Birth",  
  AGE NUMERIC LABEL="Age in AGEU at RFSTDTC",  
  AGEU CHAR(5) LABEL="Age Units",  
  ...  
);
```





Lex Jansen
Senior Consultant,
Clinical Data Strategies

Octagon Research Solutions, Inc.
585 East Swedesford Road, Suite 200
Wayne, PA 19087

ljansen@octagonresearch.com

www.octagonresearch.com

**Find this paper and more than 10,000 other
SAS papers at <http://www.lexjansen.com>**

