Paper 08

# SAS Dataset Content Conversion to CDISC Data Standards

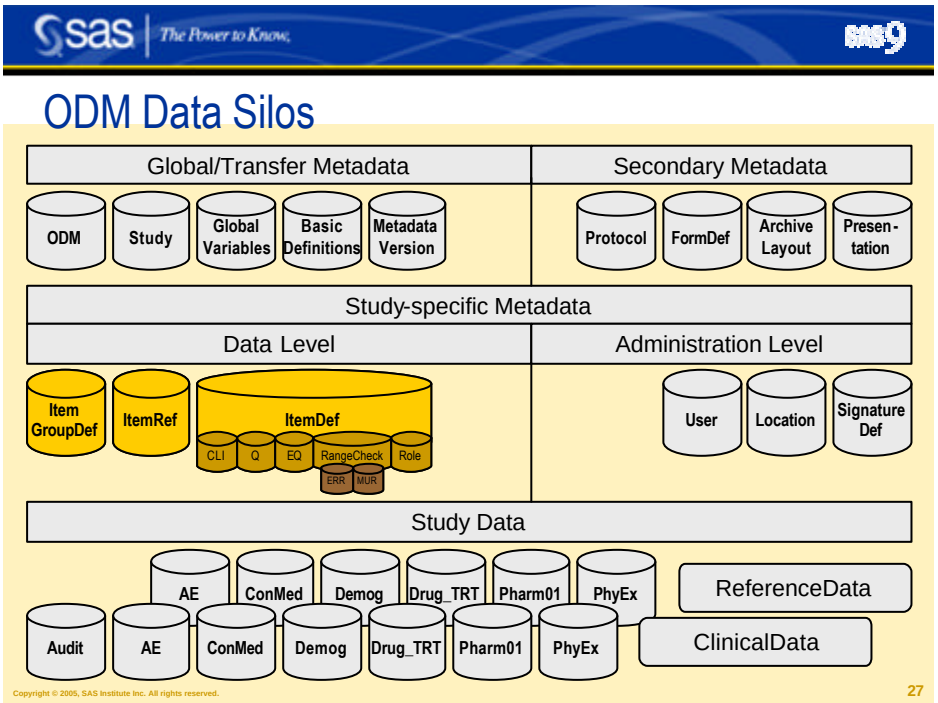
Anthony Friebel, SAS, Raleigh, NC, USA
Thomas Cox, SAS, Raleigh, NC, USA
Edward Helton, SAS, Raleigh, NC, USA

## ABSTRACT

The CDISC Operational Data Model (ODM) is a vendor neutral, platform independent format for interchange and archive of data collected in clinical trials[1]. The model represents study metadata, administrative metadata, and subject data associated with a clinical trial. Program fragments used for forward and backward conversions of data from SAS to ODM v1.2 format will be presented here, along with displays of the data before and after conversion.

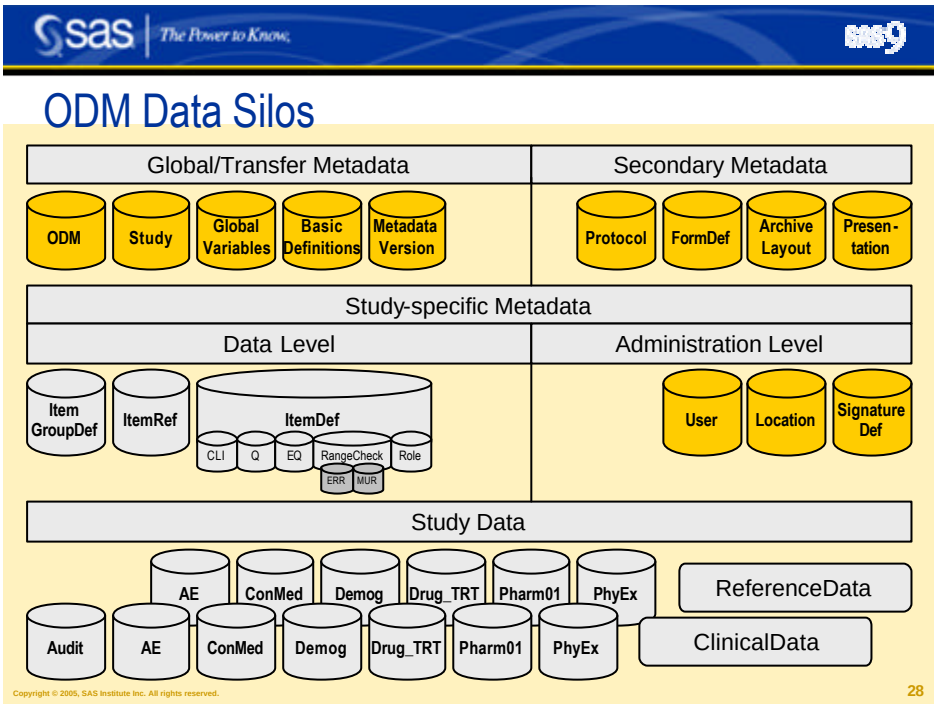
## DATA IS DATA, AND METADATA IS …

… different. The ODM provides for specification of both data and metadata pertaining to a clinical trial. It contains the fundamental metadata description for subject collected data content (tables, columns, data types, data lengths, etc.) **and** study-level metadata for the trial (participants, sites, protocol, etc.). These two forms of metadata are fundamentally different.




[1] You can learn more about the CDISC standardization efforts at http://www.cdisc.org

On the data collection side, the descriptive content translates directly to dataset construction. The names of tables, the columns contained in each table, the data type and length of each column are all transformed into tangible datasets and attributes. This is the *data transfer* side of the ODM.




On the other side lives the study metadata. While similarly constructed, these mechanisms do not contribute to fundamental analysis data from the trial. This metadata is more procedurally oriented and may take the form of tables and columns, but is provided mainly as a human-assistance mechanism within a formal definition framework. Typical components might represent a list of questions that should be asked on specific forms at particular visits during the trial. This is the *metadata transfer* side of the ODM.

Base SAS technology incorporates tools to use and report investigational data in XML conforming to the CDISC Operational Data Model (ODM) v1.2 schema[2]. Two tools are available for conversion to/from SAS and ODM format; the SAS XML Libname Engine and the new CDISC procedure. The engine may be used for *data transfer* only, i.e., content exchange consisting primarily of subject data (and the fundamental metadata for same). While the CDISC procedure may also be used for this *data transfer* exchange, it is also capable of communicating the *metadata transfer* of additional study level metadata pertaining to the clinical trial.

## ODM IN PRACTICE

Let's look at a simple read operation. This first example uses PROC CDISC to read an ODM data file called ae.xml located on our local machine, and produces a SAS dataset results.AE. The SASDatasetName option says to extract the ODM definition of AE. This can found in the ODM XML markup representation on the ItemGroupDef element where the SASDatasetName attribute option has the value of "AE".

```
SAS PROC CDISC                ODM 1.2 INPUT

libname   results 'cdisc_examples\results' ;
FILENAME  XMLINP  'cdisc_examples\ae.xml' ;
```

[2] The PROC CDISC code fragments seen here may be used in either SAS 9.1.3 or SAS 8.2 releases. SAS XML Libname Engine examples apply only to the SAS 9.1.3 release. Please refer to online documentation and support at http://support.sas.com/rnd/base/index-xml-resources.html

```
PROC CDISC        MODEL=ODM
                  READ=XMLINP
                  formatActive=YES
                  formatNoReplace=NO
                  ;

ODM               ODMVersion = "1.2"
                  ODMminimumKeyset=YES
                  ODMmaximumOIDlength=16
                  ;

CLINICALDATA      OUT = results.AE
                  SASDatasetName = "AE"
                  ;
RUN;
FILENAME  XMLINP  ;

proc contents data=results.AE varnum; run;
proc print data=results.AE; run;
```

The CONTENTS Procedure

-----Variables Ordered by Position-----

| # | Variable | Type | Len | Format | Label |
|---|---|---|---|---|---|
| 1 | __SubjectKey | Char | 16 | | |
| 2 | TAREA | Char | 4 | $TAREAF. | Therapeutic Area |
| 3 | PNO | Char | 15 | | Protocol Number |
| 4 | SCTRY | Char | 4 | $SCTRYF. | Country |
| 5 | F_STATUS | Char | 1 | $F_STATU. | Record status, 5 levels, internal use |
| 6 | LINE_NO | Num | 8 | | Line Number |
| 7 | AETERM | Char | 100 | | Conmed Indication |
| 8 | AESTMON | Char | 2 | | Start Month - Enter Two Digits 01-12 |
| 9 | AESTDAY | Char | 2 | | Start Day - Enter Two Digits 01-31 |
| 10 | AESTYR | Char | 4 | | Start Year - Enter Four Digit Year |
| 11 | AESTDT | Char | 8 | | Derived Start Date |
| 12 | AEENMON | Char | 2 | | Stop Month - Enter Two Digits 01-12 |
| 13 | AEENDAY | Char | 2 | | Stop Day - Enter Two Digits 01-31 |
| 14 | AEENYR | Char | 4 | | Stop Year - Enter Four Digit Year |
| 15 | AEENDT | Char | 8 | | Derived Stop Date |
| 16 | AESEV | Char | 1 | $AESEV. | Severity |
| 17 | AEREL | Char | 1 | $AEREL. | Relationship to study drug |
| 18 | AEOUT | Char | 1 | $AEOUT. | Outcome |
| 19 | AEACTTRT | Char | 1 | $AEACTTR. | Actions taken re study drug |
| 20 | AECONTRT | Char | 1 | $AECONTR. | Actions taken, other |

| Obs | __Subject Key | TAREA | PNO | SCTRY | F_STATUS | LINE_NO | AETERM |
|---|---|---|---|---|---|---|---|
| 1 | 001 | Oncology | 143-02 | United States | Source verified, queried | 1 | HEADACHE |
| 2 | 001 | Oncology | 143-02 | United States | Source verified, queried | 2 | CONGESTION |

| Obs | AESTMON | AESTDAY | AESTYR | AESTDT | AEENMON | AEENDAY | AEENYR | AEENDT | AESEV | AEREL |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 06 | 10 | 1999 | 19990610 | 06 | 14 | 1999 | 19990614 | Mild | None |
| 2 | 06 | 11 | 1999 | 19990611 | | | | | Mild | None |

| Obs | AEOUT | AEACTTRT | AECONTRT |
|---|---|---|---|
| 1 | Resolved, no residual effects | None | Medication required |
| 2 | Continuing | None | Medication required |

Both the SAS XML Libname Engine and PROC CDISC make use of specific attributes in the ODM markup file to identify *data transfer* content descriptions. Some of these attributes are designated as *optional* in the ODM schema description, and could be absent if a source system generating ODM XML markup was not specifically intended for SAS tool consumption. The SAS tools use the following cascade of precedence in order to determine an appropriate name

for tables,

| **ODM ITEMGROUPDEF ATTRIBUTE** | **ODM SCHEMA DESIGNATION** |
|---|---|
| SASDatasetName[3] | *optional* |
| Name | required |

for columns,

| **ODM ITEMDEF ATTRIBUTE** | **ODM SCHEMA DESIGNATION** |
|---|---|
| SASFieldName* | *optional* |
| SDSVarName* | *optional* |
| Name | required |

and for format names

| **ODM CODELIST ATTRIBUTE** | **ODM SCHEMA DESIGNATION** |
|---|---|
| SASFormatName* | *optional* |
| Name | required |

where an appropriate name is constructed, if necessary, from the ODM Name attribute by first changing any embedded blanks or other non-alphanumeric characters to underscores. All values supplied via any of the attribute values must conform to the definition of the ODM type. Failure to supply a valid name will result in a runtime error.

An application which intends to use/generate SAS names in excess of eight characters in the ODM markup, should omit the SAS-specific attribute and use the ODM Name attribute on the appropriate element.

Within XML specifics, an *optional* attribute means that the attribute and its associated value string may appear once or may be absent entirely in the actual markup. An attribute that is assigned a value of blank or NULL is present, and is not considered having been *optional.*

```
SAS XML LIBNAME ENGINE  ODM 1.2 INPUT

libname   results 'cdisc_examples\results' ;
FILENAME  XMLINP  'cdisc_examples\ae.xml' ;
Libname   XMLINP  xml xmltype=CDISCODM
                      ODMminimumKeyset=YES
                      ODMmaximumOIDlength=16 ;

data results.AE ;
set XMLINP.AE ;
run ;

proc contents data=results.AE varnum; run;
```

[3] The SASDatasetName, SASFieldName, SASFormatName, and SDSVarName attributes of the ODM are limited to eight characters in length per their original use within the SDS specification and V5 transport dataset limitations.

```
proc print data=results.AE; run;
```

```
                                The CONTENTS Procedure

                            -----Variables Ordered by Position-----

  #    Variable        Type    Len    Format       Label
  1    __SubjectKey    Char     16
  2    TAREA           Char      4    $TAREAF.     Therapeutic Area
  3    PNO             Char     15                 Protocol Number
  4    SCTRY           Char      4    $SCTRYF.     Country
  5    F_STATUS        Char      1    $F_STATU.    Record status, 5 levels, internal use
  6    LINE_NO         Num       8                 Line Number
  7    AETERM          Char    100                 Conmed Indication
  8    AESTMON         Char      2                 Start Month - Enter Two Digits 01-12
  9    AESTDAY         Char      2                 Start Day - Enter Two Digits 01-31
 10    AESTYR          Char      4                 Start Year - Enter Four Digit Year
 11    AESTDT          Char      8                 Derived Start Date
 12    AEENMON         Char      2                 Stop Month - Enter Two Digits 01-12
 13    AEENDAY         Char      2                 Stop Day - Enter Two Digits 01-31
 14    AEENYR          Char      4                 Stop Year - Enter Four Digit Year
 15    AEENDT          Char      8                 Derived Stop Date
 16    AESEV           Char      1    $AESEV.      Severity
 17    AEREL           Char      1    $AEREL.      Relationship to study drug
 18    AEOUT           Char      1    $AEOUT.      Outcome
 19    AEACTTRT        Char      1    $AEACTTR.    Actions taken re study drug
 20    AECONTRT        Char      1    $AECONTR.    Actions taken, other
```

| Obs | __Subject Key | TAREA | PNO | SCTRY | F_STATUS | LINE_NO | AETERM |
|---|---|---|---|---|---|---|---|
| 1 | 001 | Oncology | 143-02 | United States | Source verified, queried | 1 | HEADACHE |
| 2 | 001 | Oncology | 143-02 | United States | Source verified, queried | 2 | CONGESTION |

| Obs | AESTMON | AESTDAY | AESTYR | AESTDT | AEENMON | AEENDAY | AEENYR | AEENDT | AESEV | AEREL |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 06 | 10 | 1999 | 19990610 | 06 | 14 | 1999 | 19990614 | Mild | None |
| 2 | 06 | 11 | 1999 | 19990611 | | | | | Mild | None |

| Obs | AEOUT | AEACTTRT | AECONTRT |
|---|---|---|---|
| 1 | Resolved, no residual effects | None | Medication required |
| 2 | Continuing | None | Medication required |

As expected, (or not?) the results are the same. This is the value of the XML Libname Engine implementation of ODM as a *data transfer* vehicle. Although the ae.xml file contains definitions not related directly to the collected data, the libname engine sifts through the markup and successfully extracts the actual data content.

```
SAS PROC CDISC                  ODM 1.2 OUTPUT

libname   results 'cdisc_examples\results' ;
FILENAME  XMLOUT  'cdisc_examples\metadata.xml' ;

PROC CDISC        MODEL=ODM
                  WRITE= XMLOUT
                  formatActive=YES
                  formatNoReplace=NO
                  ;
```

```
ODM               ODMVersion = "1.2"
                  FileOID = "000-00-0000"
                  FileType = SNAPSHOT
                  Description = "Adverse events from the pharmaSUG file"
                  ODMminimumKeyset = YES
                  ;

STUDY             StudyOID = "STUDY.StudyOID"
                  ;

GLOBALVARIABLES   StudyName = "CDISC pharmaSUG Test Study III"
                  StudyDescription = "This file contains test data for
pharmaSUG”
                  ProtocolName = "CDISC-Protocol-00-000"
                  ;

METADATAVERSION   MetaDataVersionOID = "v1.1.0"
                  Name = "Version 1.1.0"
                  ;

CLINICALDATA      DATA = results.AE
                  DOMAIN  = "AE"
                  NAME    = "Adverse Events"
                  COMMENT = "All adverse events in this trial"
                  SASDatasetName = "AE"
              ;
RUN;
FILENAME  XMLOUT ;
```

Output of SAS data content to ODM differs slightly between the procedure and the XML Libname Engine. The ODM is very much a form-centric mechanism. Items (columns) are collected in ItemGroups (tables) which are part of FormDefinitions being filled in during StudyEvents all according to the protocol of the trial. The model derives its hierarchy from those paper forms and collections thereof.

The XML Libname Engine provides a *default hierarchy* as prescribed by the ODM and translates the physical attributes of the data source to the fundamental metadata descriptions for subject collected data content. It then produces the data content itself. This default hierarchy provides *some* of the data content access key context information.

The procedure, on the other hand, allows the user a degree of control of that same key context information. It also allows addition of some human-readable content into the generated output. Additional control statements within the procedure provide sets of parameter values that are used to populate ODM fields in the markup.

The `STUDY` and `METADATAVERSION` statements above provide key context information, while the `ODM` and `GLOBALVARIABLES` statements add human-readable content.

```
SAS XML LIBNAME ENGINE  ODM 1.2 OUTPUT

libname   results 'cdisc_examples\results' ;
FILENAME  XMLINP  'cdisc_examples\nometadata.xml' ;
Libname   XMLINP  xml xmltype=CDISCODM
                      ODMminimumKeyset=YES ;

data XMLINPUT.AE ;
set results.AE ;
run ;
```

The XML Libname Engine version of output generation is expectedly much more streamlined. We should note that the ODM SASDatasetName attribute in the markup is populated by the member name of the XML Libname Engine assigned library; in this case “AE”.

## SDTM IN PRACTICE

Whereas ODM defines both a data model *and* its expression mechanism via XML, the Study Data Tabulation Model (SDTM) is a data content only standard.

The SDTM defines a standard structure for study data tabulations that are to be submitted as part of a product application to a regulatory authority such as the United States Food and Drug Administration (FDA). The SDTM was prepared by the CDISC Submission Data Standards (SDS) Team to guide the organization, structure, and format of tabulation datasets for study data submitted to regulatory authorities. Data tabulation datasets are one of four ways to represent the human subject Case Report Tabulation (CRT) and equivalent animal data submitted to the FDA. [4]

The SDTM is composed of twenty-three(23) defined domains within six(6) broad categories. The model also provides the ability to create custom-defined domains with sets of standard variable definitions. Variables in common across domains all have similar name extensions, and the standard specifies the beginning prefix of all variables be a (typically) two-letter domain abbreviation.

It can generally be considered that an ODM content description traverses our previous diagram in a top-to-bottom fashion. The higher level study parameters dictate the execution of the forms which, in turn, define the content of data capture, which results finally in the actual structure definition of the tables and columns of the operational system to be employed.

Conversely, SDTM traverses the diagram in nearly a perfect reversed, bottom-to-top direction. All of the domain content tables, and the columns within those tables, are defined by the model with a common and consistent set of metadata descriptions applied to each of the columns within the definition. The forms aspect of the ODM is not used.

```
SAS PROC CDISC                 SDTM 3.1 VALIDATION

PROC CDISC         MODEL=SDTM
                   ;

SDTM               SDTMVersion = "3.1"
                   ;

DOMAINDATA         DATA = results.AE
                   DOMAIN = AE
                CATEGORY = EVENTS
                   ;
RUN;
```

Support for CDISC SDTM 3.1 data validation was a recent feature addition to PROC CDISC[5]. It provides data content checking against the domain definitions provided by the SDTM. The procedure currently supports 15 of the 23 SDTM domains. The Trial Design Components comprise the majority of unsupported remainder of the list.

The syntax for validation is presented above. The DATA= parameter specifies the location of your SDTM conforming data source. Values for the domain and category parameters are taken from the table as depicted below

[4] From the SDTM v1.0.0 pdf which can be found at http://www.cdisc.org/models/sds/v3.1/index.html

[5] Feature was scheduled for release in late Q1 2005.

| Supported SDTM 3.1 Domains | DOMAIN= | CATEGORY= |
|---|---|---|
| Demography | DM | Special |
| Comments | CO | Special |
| Concomitant Medications | CM | Interventions |
| Exposure | EX | Interventions |
| Substance Use | SU | Interventions |
| Adverse Events | AE | Events |
| Disposition | DS | Events |
| Medical History | MH | Events |
| ECG Test Results | EG | Findings |
| Inclusion/Exclusion Exception | IE | Findings |
| Laboratory Test Results | LB | Findings |
| Physical Examinations | PE | Findings |
| Questionnaires | QS | Findings |
| Subject Characteristics | SC | Findings |
| Vital Signs | VS | Findings |

The procedure performs the following checks on domain content of the source

Verifies that all *required* variables are present in the dataset
Reports as an error any variables in the dataset that are *not* defined in the domain
Reports a warning for any *expected* domain variables which are *not* in the dataset
Notes any *permitted* domain variables which are *not* in the dataset
Verifies that all domain variables are of the expected data type and proper length
Detects any domain variables which are assigned a controlled terminology specification by the domain and do not have a format assigned to them

The procedure also performs the following checks on domain data content of the source on a per observation basis

Verifies that all *required* variable fields do not contain missing values
Detects occurrences of *expected* variable fields that contain missing values
Detects the conformance of all ISO-8601 specification assigned values ; including date, time, datetime, duration, and interval types
Notes correctness of yes/no and yes/no/null responses

With the exception of yes/no and yes/no/null content just noted, the current procedure implementation does *not* validate the content of controlled terminology responses against a list of acceptable values. An update planned later this year will provide a mechanism for “reversing an assigned format”, but currently only the absence/presence of a format is used as the conformance indicator for controlled terminology field content.

***Implementation notes:*** The SDTM addition to PROC CDISC does NOT convert existing SDS 2.x content to SDTM 3.x representations. Neither does PROC CDISC automatically generate the V5 XPORT file from the data source.

## DEFINE.XML IN PRACTICE

The 1999 FDA electronic submission (eSub) guidance and the electronic Common Technical Document (eCTD) documents specify that a document describing the content and structure of the included data should be provided within a submission. This document is known as the Data Definition Document (e.g., "define.pdf" in the 1999 guidance). The Data Definition Document provides a list of the datasets included in the submission along with a detailed description of the contents of each dataset.

To increase the level of automation and improve the efficiency of the Regulatory Review process, the Case Report Tabulation Data Definition Specification CRT-DDS (aka define.xml) can be used to provide the Data

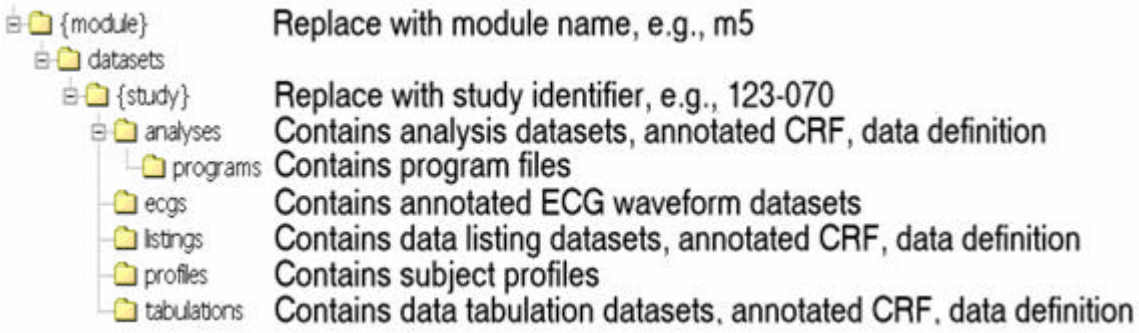
Definition Document in a machine-readable format.[6]

define.xml then, is the metadata of the SDTM tables and content rendered in a machine-readable form. But, define.xml is more than just the metadata. It also provides a structural framework for describing and referencing the collection of data tables, analysis files, annotated CRFs, program files, et.al., that typically comprise a submission.

Since define.xml is an XML-based document, it is itself fundamentally data content. The initial posting of the standard includes an eXtensible Stylesheet Language (XSL) example which can be used to visually render the content of the Data Definition (in a supporting HTML browser environment[7]) as a table of contents and a set of drill down tables representing for each of the defined component domains.

## REGULATORY CONSIDERATIONS

In July, 2004, the FDA adopted the Clinical Data Interchange Standards Consortium Study Data Tabulation Model (CDISC SDTM) for *Study Data Specifications*. These specifications are for submitting animal (SEND) and human study data (SDTM) in an electronic format. Study data typically includes information from trials submitted to the agency for evaluation and additional information to understand the data (data definition). This includes both raw and derived data.




See *Guidance to Industry: Providing Regulatory Submissions in Electronic Format General Considerations*[8] for details on creating SAS Transport and PDF files. The content of the *Study Data Specification* regarding folder specifications is that the folder also contains analysis datasets and program files. It could easily be viewed that the interoperability of the CDISC SDTM standard is required by this Version 1.0 of the *Study Data Specification.*

The analysis datasets must be supported by Version 1.0 of the ADaM General Considerations Document. Program files are already supported by the CDISC define.xml metadata specification. The required transport format for submission data is the existing SAS V5 XPORT format. It is likely to expect a near future where an XML data format consisting of define.xml metadata and extensions to the ODM data handling capability will eventually be supported as a replacement for SAS V5 XPORT format.

## CONCLUSIONS

The *Study Data Specification* allows all CDISC data standard models to converge beneath the SDTM FDA adoption. This will likely support an easier adoption and implementation in both the industry and by the FDA reviewers.

---

[6] From http://www.cdisc.org/models/def/v1.0/index.html

[7] XSL support across various web browsers varies significantly.

[8] See http://www.fda.gov/cder/regulatory/ersr/5640studydata-v1.pdf 07-21-2004 FILE FORMATS