DOI: http://dx.doi.org/10.3886/eDDIHoyle

Proceedings of the 5th Annual DDI Users Conference (EDDI13)
December 2013, Paris, France

Using Extended Attributes in Data Analysis Software

Controlled Vocabularies, Tools and DDI

Larry Hoyle[1]

Abstract

All of the major data analysis software packages now allow some form of user defined extended attributes on variables and most also allow these attributes for the datasets themselves. In each case these attributes can be seen as a pair of strings (attribute name, attribute value). They can also be seen as a subject, predicate, object triple (variable, "has" attribute name, attribute value). This paper explores potential uses of these attributes and suggests directions for developing best practice guidelines for their use.

Keywords: extended attributes, metadata, DDI, replication, reuse

1 Introduction

Many research datasets see their first instantiation in one of the major data analysis software packages, either through direct interactive entry or through being read from a text file such as a comma separated variables ("csv") file. All of the most popular packages allow for the addition of user defined attributes of variables and most allow for attaching user defined attributes to the dataset itself as well. A researcher might, for example, attach an attribute of "universe" with a value of "persons 65 or over" to a variable "PercentRetired".

This is a relatively recent and important development, greatly expanding the possibilities for reusable data and metadata. In the past, descriptive material in a dataset for a variable was limited to a text label of limited length. This label was primarily used for labeling output, and not adequate for documenting important metadata such as the question asked in a questionnaire, the universe sampled from, and units of measurement.

The unconstrained option for attribute names offers great flexibility, but will pose challenges for searches and for machine actionability in general. Imposing some structure through the use of controlled vocabularies both for attribute names and their values would increase the usability of metadata entered as extended attributes. The Data Documentation Initiative (DDI) offers one basis for a controlled vocabulary.

[1] Larry Hoyle – Institute for Policy & Social Research, University of Kansas
LarryHoyle@ku.edu

1.1 Metadata in the Research Workflow

The need for integrating documentation into the research workflow is receiving increasing attention. Long (2009) stresses the advantages of integrating documentation into early phases of the data analysis workflow. Iverson (2009) makes the case for metadata-driven survey design. Tools like Colectica (http://colectica.com/) and REDCap (http://project-redcap.org/) have been developed to facilitate capturing structured metadata early in the survey process.

Not all research data, though, is collected through surveys. Data may be "scraped" from the Web, or collected by experiment or sensors. In many of these cases the data are born in a dataset in some proprietary format, either through running some software procedure, as in collection from the Web, or typed directly into a grid in the program. Without information about the process and the individual variables, replication of the study is impossible. Recording it as soon as possible is important.

Adding structure to that information is important too. Structure facilitates retrieval and comparison across studies. Unfortunately, adding structure can create an additional burden for the researcher. Large controlled vocabularies may not be very familiar or approachable for individual researchers. Tools making vocabularies accessible through point and click may lower the barrier to their use as well as encouraging conformance with established standards. The DDI Alliance has developed controlled vocabularies which will prove useful for these tools (Data Documentation Initiative. Controlled Vocabularies).

2 Data Analysis Packages

The most popular data analysis packages handle extended attributes in different ways. A brief description of the way extended attributes are handled in each package follows. For a more complete description of all of the metadata that can be included in a dataset of each type see Hoyle and Wackerow (2011, and in preparation).

2.1 Excel with the Colectica for Excel Plugin

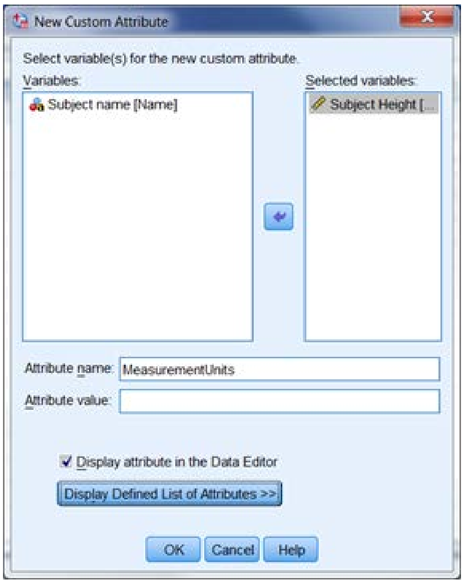
Excel does not offer a formal method for including extended attributes, but a free plug-in is available from Colectica (http://www.colectica.com/software/colecticaforexcel/download) that allows DDI based column attributes to be stored in a spreadsheet.

2.2 R

The default R Data Frame object does not have an extended attribute for variables. R does, however, allow for the assignment of arbitrary attributes to objects through the "attr" function, e.g.: `attr(rData$Fee, "MeasureMentUnits") <- "Fee is currency in Euros"`. (see http://cran.r-project.org/doc/manuals/R-intro.html#Getting-and-setting-attributes)

2.3 SPSS

SPSS manages extended attributes through the addition of “Custom Attributes” to a dataset. Arbitrary attributes may be created by selecting Data… New Custom Attribute (Figure 1). Clicking the ellipsis to the right of a custom attribute value allows it to be defined as an array of values (Figures 2 and 3). Attributes are entered in the Variable View Grid (Figure 4).



Select variable(s) for the new custom attribute.

Variables: Subject name [Name]

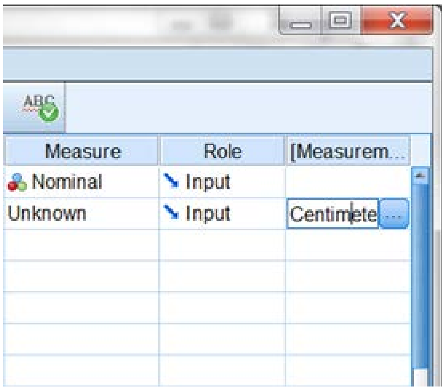
Selected variables: Subject Height [...

Attribute name: MeasurementUnits

Attribute value:

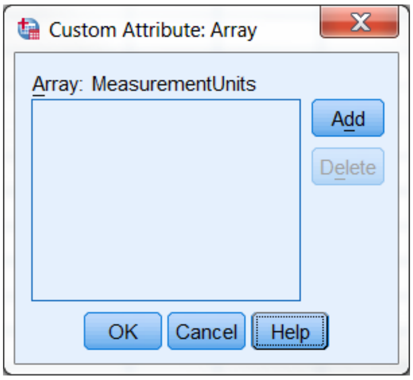
☑ Display attribute in the Data Editor

Figure 1 Creating a Custom Attribute



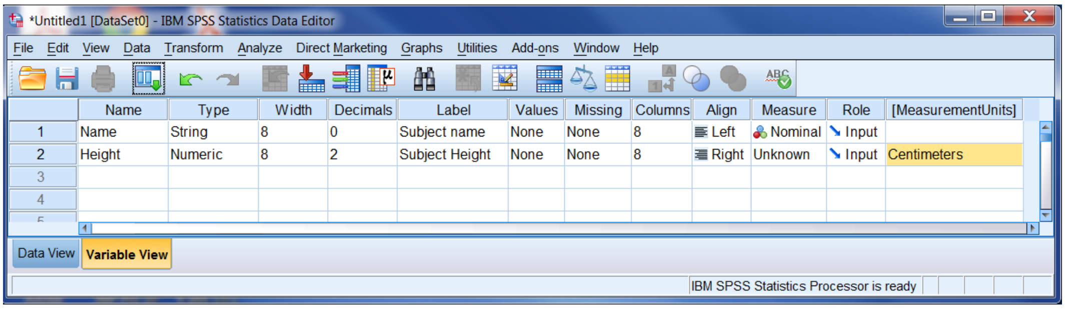
Measure	Role	[Measurem...
Nominal	Input	
Unknown	Input	Centimete ...

Figure 2 The Attribute Array Ellipsis



Array: MeasurementUnits

Figure 3 Custom Attribute Array



	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure	Role	[MeasurementUnits]
1	Name	String	8	0	Subject name	None	None	8	Left	Nominal	Input	
2	Height	Numeric	8	2	Subject Height	None	None	8	Right	Unknown	Input	Centimeters
3												
4												

Figure 4 The SPSS Variable View Grid with one Custom Attribute (MeasurementUnits)

2.4 Stata

The Stata graphical user interface (GUI) allows for attaching notes to a variable. Notes, though, are a special category of variable characteristics. New characteristics may be defined with the "char" command (Figure 5). The special variable name "_dta" refers to the whole dataset. In the example below the variable Height is assigned a "MeasurementUnits" of "Centimeters", and the dataset has a Universe of "Persons aged 65 and over".

```
. char define Height[MeasurementUnits] "Centimeters"
. char define _dta[Universe] "Persons aged 65 and over"
.  notes Height: First note on Height

. char list
  _dta[Universe]:            Persons aged 65 and over
  Height[note1]:             First note on Height
  Height[note0]:             1
  Height[MeasurementUnits]:  Centimeters
```

Figure 5 Defining and displaying Stata characteristics

2.5 SAS

SAS added extended attributes to the dataset beginning with SAS version 9.4. Attributes may be added and deleted only with the DATASETS procedure. Extended attributes may be read from the DICTIONARY.XATTRS table, from the SASHELP.VXATTR view of that table or with PROC CONTENTS. An example PROC DATASETS follows.

```
proc datasets lib=work nolist ;
   modify sales;
      xattr set ds Concept="purpose" Description="Testing Extended Attributes";
      xattr set var purchase ( Role="target" LevelOfMeasurement="nominal"
                              Description="A text description of the type of
item purchased")
                    age ( Role="reject" Minimum="0" MeasurementUnits="years")
                    income ( Role="input" LevelOfMeasurement="interval" );
```

Figure 6 Sample SAS PROC DATASETS Setting Extended Attributes

2.6 MS Access (and other relational databases)

MS Access has no intrinsic extended attribute for table columns, but a relational schema can certainly represent this type of information.

3 Controlled Vocabularies

Controlled vocabularies for extended attribute names and their possible values could be derived from the three lines of the DDI standard: DDI codebook, DDI Lifecycle, and the DDI Discovery vocabulary; as well as the sets of controlled vocabularies developed by the DDI Initiative and others. (Data Documentation Initiative. DDI Specification; and Data Documentation Initiative. Controlled Vocabularies). Table 1 shows a few possible attributes for datasets and their related elements in DDI. Complete lists are in Appendices 1 and 2.

Controlled attribute name lists like these could serve as the foundation for a set of tools allowing the collection of metadata during the normal research workflow.

Each of the attributes, in turn, may benefit from a controlled vocabulary. Measurement Units, for example, should best be described in terms of commonly adopted systems of units (e.g. see Wikipedia - International System of Units). The prototype of this application did not offer a facility for offering choices of attribute values from a controlled list.

Table 1 Three Possible Extended Attributes for Datasets

Attribute Name	DDI2.5	DDI3.1	DISCO
Abstract	stdyDscr/stdyInfo/abstract	s:StudyUnit/s:Abstract/ r:Content	dcterms:abstract
AccessRights	dataAccs	s:StudyUnit/a:Archive/ a:DefaultAccess/ a:AccessConditions	dcterms:accessRights
AlternativeTitle	stdyDscr/citation/ altTitl	s:StudyUnit/r:Citation/ r:AlternateTitle	dcterms:alternative

4 Tools

As seen above, most of the data analysis packages, with the exception of the Colectica for Excel plugin, do not offer a very user friendly user interface for managing extended attributes. None offer the capability of allowing users to choose attributes from a large controlled list. Having tools to make the process of entering more structured metadata easier could lower a barrier for researchers interested in documenting their data for reuse.

4.1 One Use Case

The prototype tool described below was developed with the individual researcher in mind, although it also could prove useful in a larger research project as well. An individual researcher might be required by a funding agency to develop a data management plan and to make the project data available at the end of the project, perhaps by submitting it to an archive. An archive might require much of the metadata described in the lists of attributes in Appendices 1 and 2.

Having structured metadata readily available might also prove useful to the researcher in several ways. Capturing the information and keeping it close to the data while it is freshly in mind should lessen the chance for error and save time overall. It should also facilitate later work, such as writing a methods section for a publication based on the data. It should greatly simplify the process of preparing a submission package for an archive. It will improve the overall quality of the dataset and enhance the prospects of reuse of the data. This reuse may even be by the original researcher. Long(2009, p.35) gives an example of a request by a reviewer for a reanalysis of the data that he was able to perform in an hour due to careful documentation that otherwise might have taken perhaps days.

4.2 A Prototype

This paper describes a prototype user interface developed for SAS.

The SAS system includes a user interface called SAS Enterprise Guide (EG), which is a .NET application implementing a graphical user interface. Enterprise Guide offers the capability to develop custom task plugins that can be used as a normal part of an analysis (Hemedinger 2012). Enterprise Guide also allows for the creation of a process flow diagram which can document (and reproduce) the entire sequence of entry, cleaning, and analysis of a dataset. The assignment of metadata to the dataset can be part of this actionable diagram (Figure 7).

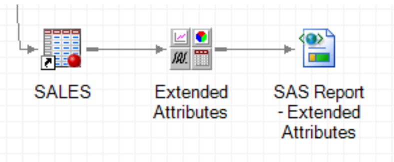


Figure 7 A Section of a Process Flow

Earlier work has also shown that metadata may be harvested from proprietary data analysis packages (Hoyle, Wackerow and Hopt 2010) and that that software metadata may be expressed in DDI (Wackerow and Hoyle 2008; Hoyle and Wackerow, 2008; Wright, 2011). This information can include categories and codes, data types, input and output formats and more. While the prototype shown here does not harvest and include this embedded metadata in DDI files, it will be a straightforward exercise to add that capability.

4.3 Alternative Tool Development Approaches

The tool described below takes an approach of using development facilities native to the SAS System. This approach could be extended by using corresponding facilities in each of the other major packages. An alternative approach would be to develop a common tool external to any one of the packages and, for example, have it generate datasets or scripts which could run in each of the packages. The latter (external) approach would have the advantage for developers of having a great deal of common code across all of the packages. It could also be implemented as a web-based service or stand-alone program. The former (native) approach would involve replicating many of the same processes in multiple languages. The native approach, however, might offer the advantage of tighter integration into each of the packages and the researcher's workflow, and therefore be less burdensome for researchers to use. Implementing native prototype applications might also encourage commercial package vendors to include the functionality in their packages.

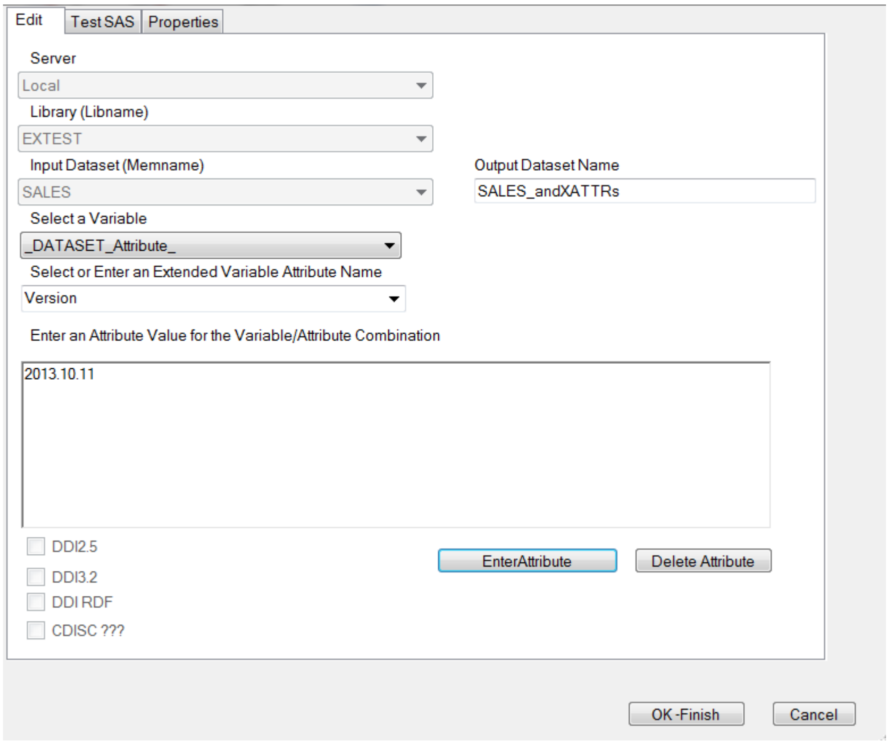
4.4 The Prototype Tool

Figure 8 shows the user interface for the prototype add-in. When the application is launched it populates the Server, Library, and Input Dataset combo boxes based on the dataset to which the task is connected in the process flow diagram (Figure 7). A Library in SAS basically corresponds to a directory or folder on the system containing the data. The application then populates the Variable combo box with the list of variables in the dataset. The application also populates the Attribute Name combo box with a list of attribute names from the appropriate controlled list. It also includes any additional user defined attribute names stored in the chosen dataset. In the background the application populates a data structure (a hash table) with any attribute name and value pairs already stored as extended attributes in the dataset.

At this point the user is ready to enter, delete, or edit name, value pairs. As the user makes selections from the Variable and Attribute Name boxes, the Attribute Value Field is filled with any corresponding attribute found in the hash table. Users may also enter their own attribute names if they are not on the controlled list. An important function of the list, though, is to encourage researchers to use common terms when available. Using the EnterAttribute and DeleteAttribute buttons performs the appropriate action on the in-memory data structure.

Since SAS can also add extended attributes applying at the dataset level, instances of structured metadata for the whole dataset, like a Data Documentation Initiative Lifecycle version (DDI-L) instance can be attached to the dataset as well. As the user selects the "dataset" variable, or an actual variable name, the list of attribute names changes appropriately (See Appendices 1 and 2 for the two lists).

When the "OK-Finish" button is clicked the application generates a PROC DATASETS (as in Figure 6) to enter or delete the extended attributes and submits it. At this point the dataset contains the extended attributes. The user can also switch to the "Test SAS" tab at any time and see the results of a PROC DATASETS run (Figure 9). The options chosen can be viewed as an XML instance at any time in the "Properties" tab (Figure 10).



Server: Local

Library (Libname): EXTEST

Input Dataset (Memname): SALES

Output Dataset Name: SALES_andXATTRs

Select a Variable: _DATASET_Attribute_

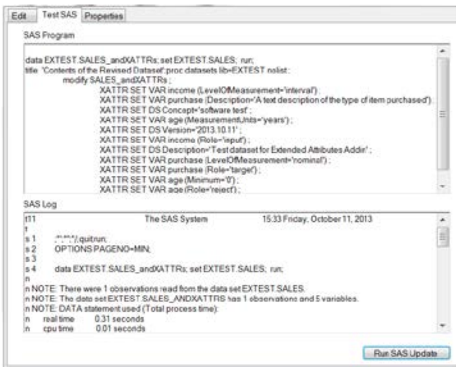
Select or Enter an Extended Variable Attribute Name: Version

Enter an Attribute Value for the Variable/Attribute Combination

2013.10.11

□ DDI2.5
□ DDI3.2
□ DDI RDF
□ CDISC ???

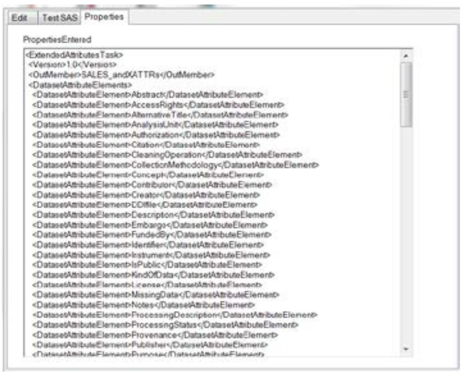
Figure 8 The User Interface for the Prototype Add-in



```
data EXTEST.SALES_andXATTRs; set EXTEST.SALES;  run;
title  'Contents of the Revised Dataset';proc datasets lib=EXTEST nolist ;
          modify SALES_andXATTRs ;
                    XATTR SET VAR income (LevelOfMeasurement='interval') ;
                    XATTR SET VAR purchase (Description='A text description of the type of item purchased') ;
                    XATTR SET DS Concept='software test' ;
                    XATTR SET VAR age (MeasurementUnits='years') ;
                    XATTR SET DS Version='2013.10.11' ;
                    XATTR SET VAR income (Role='input') ;
                    XATTR SET DS Description='Test dataset for Extended Attributes Addin' ;
                    XATTR SET VAR purchase (LevelOfMeasurement='nominal') ;
                    XATTR SET VAR purchase (Role='target') ;
                    XATTR SET VAR age (Minimum='0') ;
                    XATTR SET VAR age (Role='reject') ;
```

```
t11                          The SAS System              15:33 Friday, October 11, 2013
t
s 1       ;*';*";*/;quit;run;
s 2       OPTIONS PAGENO=MIN;
s 3
s 4       data EXTEST.SALES_andXATTRs; set EXTEST.SALES;  run;
n
n NOTE: There were 1 observations read from the data set EXTEST.SALES.
n NOTE: The data set EXTEST.SALES_ANDXATTRS has 1 observations and 5 variables.
n NOTE: DATA statement used (Total process time):
n       real time         0.31 seconds
n       cpu time          0.01 seconds
```

Figure 9 Results Running SAS



```
<ExtendedAttributesTask>
 <Version>1.0</Version>
 <OutMember>SALES_andXATTRs</OutMember>
 <DatasetAttributeElements>
  <DatasetAttributeElement>Abstract</DatasetAttributeElement>
  <DatasetAttributeElement>AccessRights</DatasetAttributeElement>
  <DatasetAttributeElement>AlternativeTitle</DatasetAttributeElement>
  <DatasetAttributeElement>AnalysisUnit</DatasetAttributeElement>
  <DatasetAttributeElement>Authorization</DatasetAttributeElement>
  <DatasetAttributeElement>Citation</DatasetAttributeElement>
  <DatasetAttributeElement>CleaningOperation</DatasetAttributeElement>
  <DatasetAttributeElement>CollectionMethodology</DatasetAttributeElement>
  <DatasetAttributeElement>Concept</DatasetAttributeElement>
  <DatasetAttributeElement>Contributor</DatasetAttributeElement>
  <DatasetAttributeElement>Creator</DatasetAttributeElement>
  <DatasetAttributeElement>DDIfile</DatasetAttributeElement>
  <DatasetAttributeElement>Description</DatasetAttributeElement>
  <DatasetAttributeElement>Embargo</DatasetAttributeElement>
  <DatasetAttributeElement>FundedBy</DatasetAttributeElement>
  <DatasetAttributeElement>Identifier</DatasetAttributeElement>
  <DatasetAttributeElement>Instrument</DatasetAttributeElement>
  <DatasetAttributeElement>IsPublic</DatasetAttributeElement>
  <DatasetAttributeElement>KindOfData</DatasetAttributeElement>
  <DatasetAttributeElement>License</DatasetAttributeElement>
  <DatasetAttributeElement>MissingData</DatasetAttributeElement>
  <DatasetAttributeElement>Notes</DatasetAttributeElement>
  <DatasetAttributeElement>ProcessingDescription</DatasetAttributeElement>
  <DatasetAttributeElement>ProcessingStatus</DatasetAttributeElement>
  <DatasetAttributeElement>Provenance</DatasetAttributeElement>
  <DatasetAttributeElement>Publisher</DatasetAttributeElement>
  <DatasetAttributeElement>Purpose</DatasetAttributeElement>
```

Figure 10 Parameters Captured in XML

5 Results and Discussion

The prototype application demonstrates that it is possible to develop an Enterprise Guide custom task that reads and allows interactive editing of the extended attributes in a SAS 9.4 dataset. This initial version raises some questions about the interface. Should there be an option to load a controlled attribute list from an external file (referenced by the project)? It may well be that different disciplines will have different needs for lists of attributes. What is the optimum size of the set of suggested attribute names? Making the lists too long may make them difficult to use.

Limiting the attachment of name, value pairs of metadata to just either variables or the dataset as a whole also does not allow for the distinction between information that applies to the instance of data or to the study as a whole. It is not clear how to handle this. Having specific attribute names for each, such as StudyPurpose and DatasetPurpose, seems overly complex.

The DDI Alliance has developed several controlled vocabularies which are relevant for some of the extended attributes like AnalysisUnit and ResponseUnit. Should controlled vocabularies for some attribute values also be presented in some fashion?

If the tool allows the option to load different lists of attribute names (and possibly corresponding attribute values), the question arises as to who would manage the lists. The DDI does have a Controlled Vocabularies Group that might address this issue for a standard DDI implementation. (Data Documentation Initiative. Controlled Vocabularies)

6 Conclusion and Future Work

Development of the prototype has shown that it is possible to develop a tool to capture a broad set of metadata embedded as a part of the data preparation and analysis workflow. Such a tool can be interactive, allowing fairly simple entry and modification of metadata, while at the same time prompting the use of controlled lists of attributes, and potentially controlled sets of values for those attributes, facilitating representation in machine actionable DDI.

Several of the questions raised in the preceding section should be addressed through testing with researchers. This kind of testing will also undoubtedly reveal other needs for the tool

An evaluation of the possibilities for native implementations for other software packages would also be useful. This should include consideration of users' working style as well as technical issues such as possible programming languages, and the development environment. Many R users, for example do not use a graphical user interface. Are there other alternatives which would better fit their preferred working style?

Figure 8 shows a list of possible output types. The custom task could be extended to output files in each of the three lines of DDI, with metadata extracted from the internal structure of the SAS file included (e.g. category and code schemes derived from user-written formats). The list of types also includes "CDISC ???". It might be worth investigating whether it would be possible or useful to include attribute names compatible with CDISC standards. CDISC SDTM, for example, specifies seven distinct metadata attributes to describe data: "Variable Name", Variable Label", "Type", Controlled Terms or Format", "Origin", "Role", and "Comments".

7 Acknowledgements

Joachim Wackerow provided valuable comments and suggestions in the development of this paper.

8 References

Data Documentation Initiative. Controlled Vocabularies | DDI - Data Documentation Initiative http://www.ddialliance.org/controlled-vocabularies [Oct 13, 2013]

Data Documentation Initiative. DDI RDF Vocabularies | DDI - Data Documentation Initiative http://www.ddialliance.org/Specification/RDF [Sept 29, 2013]

Data Documentation Initiative. DDI Specification | DDI - Data Documentation Initiative http://www.ddialliance.org/Specification/ [Sept 29, 2013]

Hemedinger, Chris (2012) Custom Tasks for SAS® Enterprise Guide® Using Microsoft .NET, Cary, NC, SAS Institute Inc.

Hoyle, Larry and Wackerow, Joachim (2011) 'DDI as a Common Format for Export and Import from Statistical Packages' European Data Documentation Initiative Conference, Gothenburg, Sweden. PowerPoint Available: http://www.ipsr.ku.edu/ksdata/DDI/ [Sept. 14, 2013]

Hoyle, Larry and Wackerow, Joachim (paper in preparation) 'DDI as a Common Format for Export and Import from Statistical Packages'

Hoyle, Larry; Wackerow, Joachim Exporting SAS Datasets to DDI 3 XML Files - Data, Metadata, and More Metadata, Paper 137-2008, SAS Global Forum 2008, San Antonio, Texas, March 2008. Available: http://www2.sas.com/proceedings/forum2008/137-2008.pdf [Oct, 13, 2013]

Hoyle, Larry and Joachim Wackerow with Oliver Hopt DDI 3: Extracting Metadata from the Data Analysis Workflow. DDI Working Paper Series, Schloss Dagstuhl, Germany, 2010. http://dx.doi.org/10.3886/DDIUseCases04

Iverson, Jeremy. Metadata-Driven Survey Design IASSIST Quarterly Spring - Summer 2009. Available: http://www.iassistdata.org/downloads/iqvol3312iverson.pdf [Oct 13, 2013]

Long, Scott. (2009) The Workflow of Data Analysis Using Stata. College Station, Tx. Stata Press

Wackerow, Joachim; Hoyle, Larry Exporting SAS Datasets to DDI 3 XML files posters presented at the IAssist 2008 conference, Stanford, CA, May 2008. (http://www.ipsr.ku.edu/ksdata/sashttp/SGF2008/)

Wikipedia - International System of Units. International System of Units - Wikipedia, the free encyclopedia http://en.wikipedia.org/wiki/International_System_of_Units [Oct 13, 2013]

Wright, Philip A. Eliminating Redundant Custom Formats, SAS Global Forum 2011. Available : http://support.sas.com/resources/papers/proceedings11/217-2011.pdf [Oct 13, 2013]

9 Appendix 1 Possible Extended Attributes for Datasets

The descriptions and comments in the following tables are derived from the Data Documentation Initiative DDI Specification documentation and the DDI RDF Vocabularies. The list is not exhaustive, but should serve as the basis for discussion.

Attribute Name	DDI2.5	DDI3.1	DISCO	Description/ Comments
Name	**fileTxt/fileName**	**l:LogicalProduct/l:LogicalProductName**		**Standard attribute in most packages**
Abstract	stdyDscr/stdyInfo/ abstract	s:StudyUnit/s:Abstract/ r:Content	dcterms: abstract	An abstract describing the study and dataset
AccessRights	dataAccs	s:StudyUnit/a:Archive/ a:DefaultAccess/ a:AccessConditions	Dcterms: accessRights	Describes access conditions and terms of use for the data
Alternative Title	stdyDscr/citation/ altTitl	s:StudyUnit/r:Citation/ r:AlternateTitle	dcterms: alternative	Any alternative title for the study or dataset
AnalysisUnit	stdyDscr/stdyInfo/sumDscr/anlyUnit	s:StudyUnit/ r:AnalysisUnit	analysisUnit	A description of the type of object studied e.g. persons
Authorization	studyAuthorization			Content, date, and authorizing agency for conducting the study.
Citation	stdyDscr/citation	pi:PhysicalInstance/ r:Citation, s:StudyUnit/r:Citation/, pi:PhysicalInstance/ r:Citation/dc:DCelements, s:StudyUnit/r:Citation/ dc:DCelements		Citation information for the study and dataset
Cleaning Operation	stdyDscr/method/ dataColl/cleanOps	s:StudyUnit/d:DataCollection/ d:ProcessingEvent/ d:CleaningOperation		A text description of the cleaning done on the data
Collection Methodology	method	s:StudyUnit/d:DataCollection/ d:Methodology	collectionMode ?	The methodology and processing involved in a data collection.
Contributor	stdyDscr/citation/ dc:contributor	pi:PhysicalInstance/ r:Citation/r:Contributor, s:StudyUnit/r:Citation/ r:Contributor	dcterms: contributor	Contributor to the creation of the dataset or study
Creator	stdyDscr/citation/ dc:creator	pi:PhysicalInstance/ r:Citation/r:Creator, s:StudyUnit/ r:Citation/r:Creator	dcterms: creator	Creator of the dataset or study
DDIfile			ddifile	A pointer to a DDI instance describing the Study or

				StudyGroup
Description	fileTxt/fileCont	l:LogicalProduct/ r:Description, p:PhysicalDataProduct/ r:Description	skos:preflabel	A general description of the dataset
Embargo	stdyDscr/dataAccs/setAvail/avlStatus	s:StudyUnit/r:Embargo	dcterms: available	Information about any period in which the data have availability restrictions
FundedBy	docDscr/docSrc/ prodStmt/fundAg	s:StudyUnit/ r:FundingInformation	fundedBy	Source of funding for the project
Identifier	stdyDscr/@ID	s:StudyUnit/@id, s:StudyUnit/UserID, pi:PhysicalInstance/@id, pi:PhysicalInstance/ UserID	dcterms:identifier	An identifier for the study or physical dataset
Instrument		d:Datacolletion/ d:Instrument	instrument	A descripton of the instrument by which the data were collected
IsPublic			isPublic	True if the data are publically available
Kind of Data		s:StudyUnit/r:KindOfData	kindOfData	The kind of data documented in the logical product(s) of a study unit. Examples include survey data, census/enumeration data, administrative data, measurement data etc.
License			license	Text of the license document for the data
MissingData	fileTxt/dataMsng			This element can be used to give general information about missing data, e.g., that missing data have been standardized across the collection, missing data are present because of merging, etc.

Notes	notes	l:LogicalProduct/r:Note		Clarifying information/annotation regarding the dataset
Processing Description	stdyDscr/method/dataProcessing	d:DataCollection/r:Description		A description of the processing don in producing the data
Processing Status	fileTxt/ProcStat	pi:PhysicalInstance/pi:GrossFileStructure/pi:ProcessingStatus		Processing status of the file. Some data producers and social science data archives employ data processing strategies that provide for release of data and documentation at various stages of processing
Provenance			Dcterms: provenance	A description of changes of ownership and custody of the dataset
Publisher	stdyDscr/citation/dc:publisher	pi:PhysicalInstance/r:Citation/r:Publisher, s:StudyUnit/r:Citation/r:Publisher	dcterms: publisher	The publisher of the dataset
Purpose		s:StudyUnit/s:Purpose/r:Content	purpose	The purpose of the study
Spatial Coverage	stdyDscr/StdyInfo/sumDscr/geoCover	l:LogicalProduct/r:Coverage/r:SpatialCoverage	dcterms: spatial	Information about the data collection's geographic coverage
Study Development	stdyDscr/studyDevelopment			Describe the process of study development as a series of development activities. These activities can be typed using a controlled vocabulary. Describe the activity, listing participants with their role and affiliation, resources used (sources of information), and the outcome of the development

				activity.
StudyGroup		s:StudyUnit/ ancestor::g:Group[1]/@id	inGroup	The group of studies to which this one belongs
Subtitle	stdyDscr/citation/ subTitl	pi:PhysicalInstance/ r:Citation/r:SubTitle, s:StudyUnit/ r:Citation/r:SubTitle	subtitle	A subtitle for the study or dataset
Temporal Coverage	stdyDscr/citation/ dc:temporal	l:LogicalProduct/r:Coverage/ r:TemporalCoverage	dcterms: termporal	The time period covered by the study
Title	stdyDscr/citation/ dc:title	pi:PhysicalInstance/ r:Citation/r:Title, s:StudyUnit/ r:Citation/r:Title	dcterms:title	A title for the study or dataset
Topical Coverage		/ddi:DDIInstance/s:StudyUnit/ r:TopicalCoverage/r:Subject		A description of the subject or topic of the study
Universe		s:StudyUnit/r:UniverseReference	universe	The set of persons, objects, or entities to which results refer
Version	stdyDscr/ @elementVersion, fileDscr/ @elementVersion	l:LogicalProduct/@version, pi:PhysicalInstance/@version		The current version of the data
Version Statement		l:LogicalProduct/ VersionRationale, pi:PhysicalInstance VersionRationale	Owl: versionInfo	Descriptive information about this version of the study or data

10 Appendix 2 Potential Extended Attributes for Variables

Attribute Name	DDI2.5	DDI3.1	DISCO	Description/ Comments
Name	**var/@name**	**l:Variable/l:VariableName**	**skos:notation**	**standard attribute in most packages**
Label	**var/labl**	**l:Variable/r:Label**	**skos: prefLabel**	**standard attribute in most packages**
dataType	**var/ @representationType**	**l:Variable/l:Representation/ l:NumericRepresentation/ @type**		**inherent attribute in all packages**
Access Level	var/security	**l:Variable/ l:EmbargoReference ???**	dcterms: accessRights	information about levels of access for this variable, e.g. public, confidential, PHI
Additivity	var/@additivity	l:Variable/l:Representation/ @additivity		e.g. ("stock" | "flow" | "non-additive" | "other")
Aggregation Method	var/@aggrMeth	l:Variable/l:Representation/ @aggregationMethod		e.g. ("sum" | "average" | "count" | "mode" | "median" | "maximum" | "minimum" | "percent" | "other")
Analysis Unit	var/AnlysUnit	l:Variable/l:AnalysisUnit	analysisUnit	information regarding whom or what the variable describes
BasedOn				Other variables on which this variable is based
Category Standard	var/stdCatgry	l:Variable/ l:ExternalCategoryRepresentation/ r:ExternalCategoryReference		Standard category codes used in the variable, like industry codes, employment codes, or social class codes
Coder Instructions	var/codInstr	l:Variable/ l:Representation/ l:CodingInstructionsReference		Any special instructions to those who converted information from one form to another for the variable
Concept	var/concept	l:Variable/l:ConceptReference	concept	The general subject to which the variable pertains
Continuous OrDiscrete	var/@intrvl	**l:Variable/ l:Representation/ l:NumericRepresentation/ l:RecommendedDataType ??**		either "Continuous" or "Discrete" depending on the range of the variable. Note that this information may be inherent in the underlying representation in the binary dataset (e.g. an integer), but a variable stored as

				type float could only have discrete values.
Derivation	var/derivation	l:Variable/ l:Representation/ l:CodingInstructionsReference, l:Variable/l:Representation/ ConcatenatedValue		a description of how the derivation was performed and the command used to generate the derived variable
Description	var/txt	l:LogicalProduct/ r:Description, p:PhysicalDataProduct/ r:Description	dcterms: description	A description of the variable
Embargo	stdyDscr/ dataAccs/ setAvail/ avlStatus	l:Variable/l:EmbargoReference	dcterms: available	Information about the period for which the variable is not publically available
Geographic Map	var/geoMap			a "URI" attribute identifying or pointing to to an external map that displays the geography in question
Identifier	var/@ID	s:StudyUnit/@id, s:StudyUnit/UserID, pi:PhysicalInstance/@id, pi:PhysicalInstance/ UserID	dcterms: identifier	A unique identifier for the variable
Imputation	var/imputation	l:Variable/l:Representation/ ImputationReference		a description of the procedure used to impute this variable
IsWeight	var/@wgt	l:Variable/@isWeight		variable functions as a weight
LevelOf Measurement	var/@nature	l:Variable/l:Representation/ l:NumericRepresentation/ @classificationLevel		e.g. ("nominal" | "ordinal" | "interval" | "ratio" | "percent" | "continuous" | "other")
Measurement Units	var/@measUnit	l:Variable/l:Representation /@measurementUnit		best taken from a controlled vocabulary such as the International System of Units cf. http://physics.nist.gov/cuu/Units/current.html
Notes	var/notes	l:Variable/r:Description		clarifying information/annotation regarding the variable
Question	var/@qstn, qstn	l:Variable/l:QuestionReference	question	a description of the question used to collect responses
Relevant Formats				Multiple formats may be relevant to a variable, but only one can be assigned to the variable at a time. This would

				allow capturing a list of formats that could be applied to the variable. (e.g. short and long value labels)
Representation	**var/ representation Type**	**l:Variable/l:Representation**	**representation**	**form of representation: codes and categories, DateTime, numeric, text (inherent in proprietary file)**
ResponseUnit	var/RespUnit	l:Variable/l:ResponseUnit		information regarding who provided the information contained within the variable, e.g., respondent, proxy, interviewer.
Scale	var/@scale	l:Variable/l:Representation/ l:NumericRepresentation/ @scale		Unit of scale, for example 'x1', 'x1000'
Universe	var/universe	l:Variable/l:UniverseReference	universe	the set of persons, objects, or entities to which results refer
Version	var/ @elementVersion	l:Variable/@version	owl: versionInfo	The version of the variable
Version Statement	var/verStmt	l:Variable/r:VersionRationale	owl: versionInfo	A text description of the current version of the variable
Weight Variable	var/@wgt-var	l:Variable/l:Representation/ l:WeightVariableReference, l:Variable/l:Representation/ l:StandardWeightReference	DescriptiveStatistics... weightedBy	variable that serves as a weight for this variable

11 Appendix 3, C# Code for the Prototype Project

An extended version of this paper with the C# code from the project is available from the author.