User's perspective: Inline documentation review and recommendations

Knutholmen Sprint | 23–27 May 2016

**Goal for the week:** To review inline documentation from a User perspective.

**Next steps:** Pass these comments on to ? for review and future incorporation. These will also form the basis of the work that will be accomplished in the Dagstuhl sprint in October.

Follow up:

Links:

## Summary:

The members of the User Perspective group present at the Knutholmen Sprint (Jared Lyle, Ron Nakao, Iris Alfredsson, and Michelle Edwards), each reviewed a section of the created XML instance for Codebook and provided either overall comments or line-by-line comments. These are copied below.

**Overall Comments/Questions:**

* We need a glossary of commonly used terms for DDI 4
* Context of the elements / attributes are required
* Concrete examples will also help
* Consistency - codelist vs code list
* Glossary - should include well worked out examples for the terms included
* Language of the documentation may be challenging to those whose first language is not english
* Documentation should be contained - link within the documents to a point in the glossary and example - no references
* Variable section - will be confusing to the user - need to find a way to clear this up
* Variable section - what will the user see? This will determine what documentation is needed
* Create a story about the codebook - using common vocabulary - think of it like an Ikea manual (e.g., to drive a car, you do not need to know how the car is assembled, thermodynamics, etc., that’s why car owner manuals focus on how the driver can do things, like open the hood, fill the gas tank, etc.)
* Two levels of documentation?

Addendum:

Highlights of our discussion on what it would take for users to move to DDI4:

* Everyone using the same version.
* Stable version (framework would be the same)
* Easier to collaborate since we’d have larger and broader support community (increases synergies of the DDI community)
* Specific functionality that folks want:
  + Support for more data types (e.g., administrative data, DH data, n-cubes, longitudinal, videos, pictures, text, etc.)
  + Support for harmonization
* Keep it simple: minimize/remove unintended complexities (as per Ornulf)
* Make life easier, then take advantage of the new functionalities as they are developed and implemented.
* Leverage and share other stuff (like tools) built on the same standard (i.e., DDI4), which decreases the hodge podge of different flavors of DDI now used out there.

## 

## 

## Jared’s comments:

This element has three concepts embedded within (statistics, weights, missing values). Can we split them out?

<xs:element name="Statistic" type="StatisticType">

<xs:annotation>

<xs:documentation>The value of the statistics and whether it is weighted and/or includes missing values.</xs:documentation>

</xs:annotation>

</xs:element>

These two elements aren’t readily understandable to an external user. It would help to have a glossary of terms to explain terms like MeasureRole, ViewPoint, etc.

<xs:element name="AttributeRole" type="AttributeRoleType">

<xs:annotation>

<xs:documentation>A MeasureRole identifies an InstanceVariable as being an attribute within a ViewPoint. </xs:documentation>

</xs:annotation>

</xs:element>

<xs:element name="Concept" type="ConceptType">

<xs:annotation>

<xs:documentation>Unit of thought differentiated by characteristics [GSIM 1.1]</xs:documentation>

</xs:annotation>

</xs:element>

## Michelle comments:

Several elements use names of other elements in their definition, for example:

|  |  |
| --- | --- |
| Quality Statement | A statement of quality which may be related  to an external standard or contain a simple statement of internal quality goals or expectations. When relating to an external standard information on compliance may be added providing a reference to a ComplianceConcept, an ExternalComplianceCode, as well as a description. Optionally, a general statement of quality may be provided using OtherQualityStatement |

Typos:

|  |  |
| --- | --- |
| RectangularLayoutType - IsDelimited | Need to correct IsFixedWidth |

|  |  |
| --- | --- |
| TemporalCoverage | Describes the date or time period covered by  the annotated object. Allows for the use of specifying the type of coverage date as well as associated subjects or keywords |
| Subject | A subject that describes the topical coverage of the  content of the annotated object. Subjects are members of structured classification systems such as formal subject headings in libraries. Uses an InternationalCodeValue and may indicate the language of the code used |
| Keyword | A keyword that describes the topical coverage of the  content of the annotated object. Keywords may be structured (e.g. TheSoz thesauri) or unstructured and reflect the terminology found in the document and other related (broader or similar) terms. Uses an InternationalCodeValue and may indicate the language of the code used |
| HasPrimaryContent | A whitespace-delimited list of the DDI URN identifiers  of the objects contained in an XML instance or RDF graph which could be considered the primary objects or entry points. In a Codebook View, for example, the top-level Codebook object. In some views, such as the Agents View, there my be more than one primary object (Individuals, Organizations, and Machines in this case) |

## Ron’ Element Documentation Comments

## (he reviewed elements: AgentAssociationType to Paired External Controlled Vocabulary Entry Type.)

AgentAssociationType:

“The role of the of the Agent within…”

=> typo. delete “of the”

BasedOnRationaleCode:

“RationaleCode is primarily for internal processing flags within an organization or system. Supports the use of an external controlled vocabulary.”

=> Can we provide links to where other elements are defined when they are used in documentation? (e.g., RationaleCode used in BasedOnRationaleCode)

Commonality:

A description of the common features of the two items using a StructuredString to support multiple language versions of the same content as well as optional formatting of the content.

=> another example of referencing another element within its documentation: what is StructuredString? possible solutions: link references of element names to their documentation, and/or create a glossary of elements with definitions with linking between the glossary element (or other terms that require definition) and where it is used

Command:

“Provides the following information on the command: The content of the…”

=> avoid use of the element in the documentation/definition.

DynamicText:

Structure supporting the use of dynamic text, where portions of the textual content change depending on external information (pre-loaded data, response to an earlier query, environmental situations, etc.)

=> another example of using a term for its documentation that is the element. (what is dynamic text?)

CommandFile:

“Identifies and provides a link to an external copy of the command, for example, a SAS Command Code script.”

vs.

CommandFile:

"Identifies and provides a link to an external copy of the command, for example, a SAS Command Code script. Designates the programming language of the command file, designates input and output parameters, binding information between input and output parameters, a description of the location of the file , and a URN or URL for the command file”

=> check that same core text is used when the same element is documented. CommandFile had two places that i saw documentation text with the 2nd one having additional text from the 1st. however, if the context of the documentation for the same element requires extra contextual text, then it may be okay (?)

TypeOfIndividualName;

The type of individual name provided. the use of a controlled vocabulary is strongly recommended. At minimum this should include, e.g. PreviousFormalName, Nickname (or CommonName), Other.

=> reference to “controlled vocabulary” (perhaps link to the glossary definition?). links to other element names referenced?

LocationName

Name of the location using the DDI Name structure and the ability to add an effective date.

=> reference to “DDI Name structure” (perhaps link to the glossary definition? should be “Structure”?).

DateContent:

This is the date expressed in a non-ISO compliant structure. Primarily used to retain legacy content or to express non-Gregorian calender dates.

=> typo. calendar to “calendar”

PairedExternalControlledVocabularyEntryType:

A tightly bound pair of items from an external controlled vocabulary. The extent property describes the extent to which the parent term applies for the specific case.

=> another example of linking to glossary for definition of “controlled vocabulary”(?)

ControlledVocabularyID:

The ID of the code list (controlled vocabulary) that the content was taken from.

ControlledVocabularyName;

The name of the code list.

ControlledVocabularyAgencyName:

The name of the agency maintaining the code list.

ControlledVocabularyVersionID:

The version number of the code list (default is 1.0).

OtherValue:

If the value of the string is "Other" or the equivalent from the codelist, this attribute can provide a more specific value not found in the codelist.

ControlledVocabularyURN:

The URN of the codelist.

ControlledVocabularySchemeURN:

If maintained within a scheme, the URN of the scheme containing the codelist.

=> the preceding five elements (ControlledVocabularyID - ControlledVocabularySchemeURN) used code list and codelist. if they are correctly used, then never mind, otherwise need to be consistent. perhaps other examples of this in documentation??

**Iris Alfredsson comments**

In “my” section I found a number of elements that needs a richer description and/or examples:

**DataPoint**

A DataPoint is a container for a Datum.

**DatapointOrderedPair**

The pair of Datapoints in a record which are being placed in a sequence.

**DataRecord**

A Record is a Collection of DataPoints with an optional OrderRelation.

**DataStoreSummary**

Holds information about the substantive contents of the DataSet, e.g. case-quantity/record count.

**Datum**

A Datum is the designation of a concept with a notion of equality defined

**IdentifierRole**

An IdentifierRole identifies an InstanceVariable as being an identifier within a ViewPoint.

**ImplementedInstrument**

ImplementedInstruments are mode and/or unit specific.

**InstanceVariable**

The use of a Represented Variable within a Data Set.

**MeasureRole**

A MeasureRole identifies an InstanceVariable as being a measure within a ViewPoint.