Expert Committee Meeting

October 18-20
Ann Arbor, MI

Present:
Mark Diggory, Pascal Heus, J. Gager, Arofan Gregory, Sanda Ionescu, Jeremy Iverson, I-Lin Kuo, Ken Miller, Tom Piazza, Jostein Ryssevik, Wendy Thomas, Mary Vardigan, Joachim Wackerow, Oliver Watteler

Tuesday, October 18

Moderator: Arofan Gregory

Wendy reported on progress made so far, and presented goals for the current meeting. The timeline laid out at the Edinburgh meeting in May 2005 has been met, with all the working groups having already submitted proposals to the SRG. The goal remains to move Version 3.0 to public review by June 1, 2006.

The present meeting will review all of the proposals submitted to this date. It will also discuss a number of issues critical for drafting the DDI 3.0 schema, like module and instance versioning, external and internal pointers, IDs, treatment of language information, and controlled vocabularies. A working plan will be set up for creating the necessary accompanying tools and materials--Tag Library and Best Practices Recommendations, tools for converting Version 2.0 instances to Version 3.0, tools for creating Version 3.0 markup, etc. We will also need to consider which development goals need to be met in Version 3.0 and which can be postponed for Version 3.1 or later.

Time

J Gager presents the Time Class proposal, submitted by the Aggregate Data, Time and Geography (DDI-ADG) working group.

The proposal includes a Temporal Coverage Class designed to describe the temporal dimension(s) of the data , and a series of administrative "date stamps" for operations like data collection, processing, and archiving.

It was pointed out that under Temporal Coverage, we need to clarify what "reporting date" stands for. It would be used when data are only reported at fixed intervals, while being continuously collected. It was also noticed that, in the administrative dates area, many steps are missed -- in study and questionnaire design, pilot testing, data entry, etc. It was generally agreed that we need to identify a typology of datable events throughout the study life-cycle and then enable a more generic date type that can be applied to each of them.

The observation was made that while the proposal covers study-level temporal information, it overlooks the description of time variables--how do we best describe these to adequately capture time-related information? It was agreed that the SRG will take a closer look at this, with input from ADG members.

Geography

J Gager presented the Geography Coverage Proposal, also submitted by the DDI-ADG working group.

The desirability of making the Geographic Bounding Box repeatable wasdiscussed. A repeatable Bounding Box may help describe with increased precision separate geographies that are covered in the same data file. On the other hand, by making this class repeatable we would be changing its very definition, which implies total coverage. SRG may consider adding an exclusion attribute to the G-Ring class, for cases where it iseasier to define the excluded area.

It is also deemed that we need more information than just the URI about the external authorities we would be referencing--we would want to specify version, for instance, and maybe other things.

Aggregate Data

J Gager presented the Aggregate Data Proposal, also on behalf of the DDI-ADG working group.

The proposal includes three modules for markup of aggregate data, describing:

1. Cube and noncube data in an external fixed-format or delimited array data file (similar to the structure that is part of Version 2.0)
2. Cube data in which both the metadata and data reside in an external file (e.g., spreadsheet)
3. All values of the cube, including coordinates, measure, and attributes described in the instance; data are inline

The proposal enables attaching attributes (typed information) to both entire cubes and individual cells. Also mentioned was the need for attaching metadata to cube regions that may share certain characteristics, as opposed to the entire cube, or a single cell. To that end, we need to be able to define a cube region, as an intermediate level between cube and cell. A region could be defined by specifying subsets of dimension values. It is probably better to enable a choice to define such a region by either inclusion or exclusion--whichever is more convenient.

The issue of display was also raised. Do we provide for a "recommended" display, or a default presentation of the cubes? Presentation is usually not part of the metadata, although It was pointed out that there are some data distributors (Statistics Netherlands) who use software that controls display.

J will add ID values to Measure. He also asked the group to give further consideration to the attributes already associated with the element Measure, which may become redundant when we introduce the new Attribute class.

Instrument Documentation

Tom and Jeremy presented the Instrument Documentation Proposal, on behalf of the Instrument Documentation Working Group.

Gina-Qian Cheung and Karl Dinkelmann from SRO (ISR) were invited to attend the presentation and discussion of this proposal.

The proposed model focused on capturing and rendering in DDI the output from existing CA interviewing tools. How will the existence of a separate data collection module affect the variable description as it currently stands in Version 2.0? Will the variable only link to the question spelled out in the Instrument section? It may be best if we retain a placeholder for the textual question at variable level, as well as the possibility to link back to a question in the Instrument. But this kind of double iteration may also pose processing problems. A possible solution may be to allow question text in the variable description only if there is no instrument module, or the module does not contain the question, and only allow a reference to the question if it is included in the instrument section.

Referring to the Response Domain class, it was pointed out that language also needs to be documented, as well as response attributes (conditions).

Some suggestions were presented on how to document multi-language instruments, and a markup example was shown to demonstrate its feasibility. In this model:

1. A module is a list of a single type of element, or a collection of related lists. So, for example, a module may be a list of questions, a list of variables, a list of categories, or a collection of the three above lists.
2. Elements within the module are identified by an ID attribute. (Though this is really a key rather than an ID).
3. Multiple variations of a module may exist. Corresponding elements in the variations share the same ID. This allows the referring entity to use the same ID, and be ignorant of the actual module.
4. Resolution of linking is done in a three-step process:
   - resolving the document location, possibly via a registry
   - retrieving the document and selecting the appropriate variation of the module, based on module-level attributes such as xml:lang, ddi:file, etc. The actual selection of which module is controlled by the application.

    - retrieving the appropriate element within the module via ID
5. Text which is "translateable" should be placed in element content as much as possible. 6. Text which is machine-actionable" and not translateable should be placed in attributes. 7. Text which serves a dual purpose -- machine-actionable and translateable -- should be separated into two pieces and handled separately.

A translation would be done by taking a reference module and asking the translator to translate all the text between the tags, leaving the attribute text alone.

This proposal involves separating questions from their response domain. If the response domain were separate, it could be applied to all questions that would share it.

Comparative Data

Oliver presented the proposal package submitted by the Comparative Data working group.

The first part of the proposal reviews concerns and goals in comparing datasets. The package also contains two more concrete proposals, drafted by Reto and Achim, that cover some of the aspects involved in the comparison effort.

Since the comparison issue is so complex, Arofan suggested that we focus the discussion on those goals that we can actually achieve in the given timeframe. We should definitely concentrate on variable-level comparison, and perhaps also explore how, and to what extent, we should document methodology for the purpose of comparability. Even if the most pressing demands are for comparing variables, we also need a comparison mechanism at study level.

Will we limit Version 3.0 to comparing studies that are comparable by design? Studies that are not comparable by design need to be harmonized, so then we would also need to document the harmonization process. That could be done, even in Version 3.0.

In Version 3.0 we can enable comparison at the level of variables, questions, and concepts. We will provide a way of mapping relationships. Arofan suggests that for comparison purposes we create a registry where we would store variable descriptions and their relationships; we could then compare datasets by querying the registry (by variable, question, concept), rather than doing multiple dataset to dataset comparisons.

The standards for interfacing with the registry would be built into the DDI schema. As an alternative to querying a registry, DDI will also enable pairwise comparisons.

Wednesday, October 19

Moderator: Wendy Thomas

Arofan summarized yesterday's discussion around the creation of a registry for comparing studies at variable level. Maintenance-related concerned were raised, as well as the possibility of creating hierarchies for comparison in a local registry.

Reto Hadorn was contacted for a telephone conference call. The discussion centered on the idea of a registry, which may be similar in concept to Reto's projected *standard* study that would be used to compare datasets.

In addition to facilitating comparisons, such a registry would also be a valuable resource in new study design, being used as a concept/variable/question bank. It could also be used to store and provide information about translations of variables and questions.

The group also needs to explore the possibility of making the creation of a registry a separate project from DDI v3.0 so that it does not delay the launching of the new version. Version 3.0 will definitely support pairwise comparisons, and an interface with a registry may be enabled in Version 3.1

Implementation might not be too difficult, as there are open-source registries that could be used with some modifications. Intellectual property issues will be unlikely as long as we are not changing anything from the initial product.

Achim presented his proposal for handling comparative data. In the discussion that followed, it was suggested that Derivation Command be made repeatable. Achim's proposal requires this element to be typed in a structured language that is not software-specific but allows machine-actionability. It was pointed out that in the Derivation class we also need to specify the input variable.

Regarding Achim's idea to allow further subdivisions in the case of large modules, there was a warning that such divisions may create problems, for instance in areas like referencing and validation.

Longitudinal Studies

Sanda presented her proposal for marking up longitudinal data in DDI. Many of the issues involved in longitudinal data are addressed by the comparative data proposal.A discussion ensued regarding classification by concepts. Will we include concepts in the DDI or will they sit outside the DDI, in a scheme that will be referenced? The problem with bringing external classifications into the DDI is that one then needs to manage versioning.

Complex Files

The Complex Files Proposal, submitted at the DDI Alliance meeting in May 2004, was also briefly reviewed. In addition to describing "simple" but related data files with a view to merging them for analysis, should we also provide for the description of complex datasets that are being distributed as such, and may necessitate splitting into smaller units?

A"wish list" for DDI Version 3.0 includes revising the terminology so that it no longer appears as survey-biased, creating a glossary, and enabling better control of formatting text.

Agents and Agencies

Pascal presented some ideas on identifying agencies within the next version of the DDI. One option might involve creating and maintaining an "address book"--or an agencies registry--to which we could map all the various roles involved in the study life cycle. Maintaining such a registry could prove difficult. Linking people and organizations within a hypothetical registry might also prove problematic. Individuals may belong to several organizations, or may change affiliation, and this kind of information needs to be constantly updated.

Roles

Ken has compiled a list of roles that agents or agencies can play throughout a study's life cycle. Examples are Principal Investigator, Sponsor (Funding Agency), Producer, Depositor, Copyright Holder, Original Archive, Distributor, Publisher, etc. Roles played within the archive: data processing, technical support, user support, translation, etc. Roles could be included in a typology. They would link to agencies, and would appear where applicable in the study's life cycle. They could also be used to indicate the "source" of metadata relating to stages in the life cycle.

Translations

Ken also reviewed issues related to translations/multilingual studies. If a study is multilingual by design (applied to different populations), translations (primarily of questionnaire items) occur in the preparation stages. Translations can also occur post-production, mainly for the purpose of data definition and discovery. We need to document when the translation occurred, the original language, the translation language, who was the translator (with agency affiliation?) as well as his/her nationality, and perhaps the purpose of the translation. We need to allow multiple entries for translation language, but also for the original (some countries have two official languages, e.g., Canada). We may also want to document whether the translation is just textual, or it includes cultural/contextual adaptations (this is an important distinction for comparability).

If translations become new DDI instances, they need to point to the original. Likewise, translated modules (when only some modules, and not the whole instance, are being translated) will need to point to the originals. Also, if there are multiple originals, we need to specify which original was used in a particular translation.

How do we document translations? We could have typed translation notes at all major levels, or log translation event(s) and document what was translated. There is general agreement that the best solution is to have translation documented in a special module and then point to it.

The issue of unique identifiers was raised--will translation of the same variable have different IDs? It is possible to guarantee uniqueness with an ID- plus-language combination.

It was suggested that even with the new modular approach it might be advisable to keep at least the microdata variable description as it is right now, this being a format that most people are used to, and is more easily approachable and convenient to create out of existing source documents (statistical syntax, data dictionaries, etc.).

Thursday, October 20

Moderator: Jostein Ryssevik

Topics discussed: Versioning, Extension Mechanisms, Referencing, Grouping, and Reusable Classes, Classification Schemes, Tools, Workplan, and Timeline.

Versioning

Wendy presented a summary of current discussions and decisions on versioning.

We will be versioning the content of instances, not the instances per se. Any change in content (major or minor) will mandate a new version, unless we are creating a new object (we will have to be careful in defining what kind of changes produce a new version vs. a new object).

Versioning will only apply to published items (not internal updates) and will be done at the module level.

It was added that we will also need to be able to document versioning of the data (besides metadata).

The IDs will be URNs. At the top level there will be a URN--or a combination of ID and Agency, to disambiguate between IDs. This will be necessary for external referencing. It was recommended that we use the domain name as Agency. The topmost level will also include placeholders (paths) for the various modules/sections, indicating where to find a part in the system.

Referencing

It has generally been agreed that referencing will be done upward, from lower level modules to groups and wrapper. Will we need to rethink any of this? There was a warning that upward referencing may pose problems for applying inheritance, where properties of the group are inherited down to lower level modules.

Grouping

The way we design the grouping mechanism is also relevant for inheritance, which is linear and does not work across hierarchies. Grouping will be applicable at any level.

The group is currently seen as containing all the similarities, but we are not quite sure yet how, and at what level, we will document the differences. In any case, users will also have the option of repeating everything (documenting both similarities and differences) in each Study Item.

It was acknowledged that there may be limitations to our current grouping approach, but they will be easier to detect when we actually start testing the new specification.

It was pointed out that we also need to define more precisely what constitutes a "study". That becomes even harder with aggregate data, where it seems more appropriate for cubes to reside at a lower level than dimensions, while the latter might be described at group level.

Reusable classes

Reusable classes currently cover agency, action log, translation log, notes, time, geography, and other materials. Type lists need to be drafted for action and translation logs, other materials, and notes. The SRG also needs to review, and amend if necessary, reusable classes that were not addressed by working groups.

Classifications

Classifications/controlled vocabularies can be handled in several ways:

- DDI provides a typology but allows local additions (for example, a controlled list that also includes the option "other" as a placeholder for user-created lists).
- DDI provides a mechanism to link to external typologies.
- DDI enables references to an external typology with local modifications (these are generally restrictions allowing linking only to a part of a list).

External typologies might not be in DDI, but it would be preferable to have them in DDI format--this will make them machine-actionable (different formats are harder to process).

We will need a DDI standard for creating classifications. Such a standard would be useful in classifying categories, variables, and questions. If the same categories/category groups--or questions, or variables--were used repeatedly, we could classify and externalize them and then just point to them. (However, it was noted that this approach would break the current data description layout).

We also need to put together a list of all the places in DDI where we need controlled vocabularies.

Extensions; Modularity

A good approach to extensions would be to allow an additional key-value pair of attributes (name and value of key) on each repeatable element. These attributes would be ignored by general applications, and taken into consideration by local ones.

We do want to provide clear rules for extensions, as part of the standard. Processing instructions are allowed in xml, and we might want to include them in our specification. They would indicate a generic way to make changes. (The downside is that many Web services strip these out.)

Mark reviewed the modularization and extension mechanisms that are available in the W3C schema. Modules can be: imported (module keeps its namespace), included (within the namespace of parent schema), included with redefinition of included elements and attributes, or mapped with xsi:any or xsi:type.

It was explained that we do not want to use xsi:any mapping, since it will result in non-operability. Likewise, we do not want to use redefinition.

Importing extensions with their own namespace seems the most satisfactory mechanism; elements' names can be the same, but the different namespace will indicate that they are being used differently.

If we use namespaces to mark changes to the standard, these changes will become transparent at all times. A conformance checker applied to the DDI when it enters and exits an organization will thus easily identify all changes to the standard.

Should we just have just one DDI namespace, or should we have a namespace for each module? Multiple namespaces are harder to manage, but we might want to try them first, and see how they work.

It was suggested that we change the word "codebook" in the docType to indicate that we are distancing ourselves from the old printed codebook format.

Tools

Pascal made a brief presentation of the Microdata Management Toolkit developed by the World Bank in cooperation with Nesstar. The toolkit includes an improved version of Nesstar Publisher that integrates data, metadata, and links to external resources, has a quality control check, and a project organizer. The integrated file--which is DDI compliant--is then fed into a CD-ROM builder that allows product customization. Data can be included or excluded, and a PDF version of the documentation can be produced. The CD-ROM builder works with both NSDstat and DDI formats, and will be an open source when released.

Plans were then discussed for providing some authoring tools for DDI 3.0 as well as a conversion tool that will translate Version 2.0 instances into Version 3.0. We might want to facilitate Version 3.0 to Version 2.0 conversions, too, but that is not so urgent.

The first tools we provide will be intended for testing Version 3.0. Later, these could be improved to build a DDI starter kit.

DDI Version 3.0 will be too complex to use with a simple xml editor. We could create an authoring tool using Xforms, which bind html forms directly to an xml instance. It was suggested that we also provide a simple conversion tool that will translate statistical syntax (the most commonly used is SPSS) into DDI 3.0.

We also need to provide good documentation and training materials, including use case examples. Standard documentation will be included in the schema itself.

Next year's IASSIST conference is a good venue for presenting the new DDI 3.0, possibly including a double DDI workshop.

Working Plan and Timeline

We are generally doing well, with ample time left for putting out a draft of the new version, internal and public testing, and final release by the end of next year.

Specific tasks and deadlines are being reviewed; they will be included in a spreadsheet and posted on the Web.