## Discussion on Roadmap and design rules

* DDI as a suite of standards rather than unification to a single “DDI”
  + There is a single “DDI” concept as an approach but different products may have differing coverage within the DDI conceptual area and/or different structures for expressing those
  + The user should use the product that addresses their needs
  + There is not the assumption that one should/must move from one DDI product to another except to access different functionality
  + Implications for design rules, coverage, definition of DDI product in the suite
    - We need clear mappings of overall DDI conceptual area and what each product addresses
    - Agree that a product is not prevented from expanding its DDI coverage if needed by the community of use
    - Coverage of corresponding areas of the DDI by different products should be similar (i.e. have the same conceptual basis even if expressed differently)
    - Redefine the design/development rules of individual products to reflect this change in conceptualization of products
* Clearly define the coverage and application areas/restrictions of each product
  + Better definition of the audience
    - Decision makers – High level decision makers. General high level functionality. What the standard does and how it works with others (marketing world)
    - Content providers – information on how objects work together (Training world)
    - Developers who are focused on the model and how to implement use of the standard (technical world)
  + Relate coverage to the DDI conceptual model
    - We need to express the conceptual (to do)
* Conceptual level mapping i.e. what DDI covers and how it is expressed in different products
  + Use to identify commonalities, points of similarity/dissimilarity
  + Some of these need to be bullet points for Marketing and Training to explain DDI overall and applications for specific products
    - Codebook is after-the-fact and Lifecycle is before/during-the-fact
    - Codebook is specifically about a “study”, Lifecycle can be used to support non-study metadata/data (Question banks, cross-study classifications or concept structures)
    - The marketing documentation should focus on the intended best use (happy-path) of each product

This week:

* Work on the documentation
  + middle section (content providers) how things work together (best practices)
  + definition improvement
* Creating the conceptual model
* Map at least Codebook and Lifecycle to coverage of conceptual model
* Application areas
* Organize the content we do have to this type of presentation model and see where there are gaps to fill in

COGS

* What is COGS and what is it good for? We need to clearly describe this so we can see where we may need other change

In terms of DDI4:

* Review features of UML that are being used now, or may be used and determine the role they play in the model and resultant XMI
* Composition relationships – can this be handled by the target type (non-identified with relationships)
* How are the specialized relationships, such as refine and trace, being used? As a specialized set of XMI or as part of the release of the model.
* Graphs are generated by COGS not directly by the canonical xmi (canonical xmi does not have diagrams)
* What can be managed by modeling rules applied by COGS
* COGS currently puts out the EA flavor of normative OMG UML 2.4.2 and 2.5 with diagrams
* Should DDI4 be using COGS even just for documentation
* Maintenance of DDI4 over time – is there an alternative maintenance structure like EA repository

Codebook:

* Minimum get the documentation into COGS to produce standard documentation
  + Look where Lifecycle URN is currently available
* Over the next year look at this in terms of the review of design rules and product specification

Lifecycle

* #%& privacy attribute
* Substitution groups for Physical Structure – needs to be remodeled
* Everything by reference (dropped inline options)
* Dropped specific schemas
  + How does typeOfXXX correspond to specific group types?
* So a 3.4 would be a technical structure change incorporating above
* Which outputs do we want to have published? For 3.4?
  + JSON Graph
  + XMI
  + XML
  + RDF
* Add some more of the broad documentation
* Determine what Topics we want
* Directions on how to make or submit a patch in the Git Repository
  + Where do want patches and issues to be filed?
* Very that what’s coming out of COGS is not contradictory to what’s coming out of the DocFlex
* Get all documentation production as part of the build not a separate hand processed event

DDI Resolution

If we have the fixes entered for 3.3 we could start on a prototype on it. Create the prototype and then share in the community for further development and finalization.

URL format for API resolution (resolving a DDI URN to an API format)

Best Practices Document

* Default values for content
* Subtypes “TypeOfXxxx” where this may be used in future. How to publish organizational Controlled Vocabularies.
* Clarify what should be in this document and what should be in standard documentation.
* Questions – minimum questionnaire package
* Conceptual framework – structure, content