Where DDI4 is today:

# Production:

XMI: We can produce an XMI from Drupal that captures what we currently capture in Drupal

XML: We can produce a valid XML schema for the DDI4 library and Functional Views

RDF: We cannot currently produce RDF and this is primarily because we have not determined what the RDF will consist of. During the last Dagstuhl Sprint there was some consensus among those in the working group that we should incorporate current vocabularies directly into DDI. This decision has not been vetting in the broader DDI community and the ramifications have not been broadly discussed. There are advantages and disadvantages to this approach which have not been fully explored. In the past we have run into problem when incorporating external vocabularies into DDI (XHTML and Dublin Core). The alternative of creating a DDI vocabulary and mapping it to existing vocabularies is another possibility that has not been fully discussed. A mix between the two (using existing vocabularies for simple structures and DDI for more complex. This topic has not been discussed in the Modeling Team.

Drupal: Drupal does not currently capture all that we need to in terms of documentation. There is a list of changes that have been requested and placed in a workpackage for an external programmer. Even with these changes there are problems in terms of tracking changes to the content. It has supported a wider range of individual’s involvement and this is one of its big payoffs. Movement from Drupal should only take place if we can show another approach retains the wins and deals with some of the problem issues with Drupal. EI continues to have the issues of high learning curve, inconsistency between versions, and difficulty in extracting documentation (at least these were the initial reasons for moving away from this approach).

Production Process (automation): We are not there yet. As always, we are close. Issues mostly seem related to individuals time to focus on and resolve the problems.

# Coverage:

DDI4 deals with complex data types like DDI 3.2 and is not a problem. The style of XML is very different and there is no desire to replicate the style and serialization approach of DDI 3.2.

StudyUnit: Classes used by a simple codebook have been incarnated into DDI4 but strictly in the context of a simple codebook. In short, these have not been seriously reviewed in terms of a “publication” of a study, project, or other document intended for persistence, preservation, etc.

Group: The features of Group in terms of reusable variables, questions etc. is being addressed in a different way primarily through the variable cascade and possible question/capture cascade. This work is tentative and has not been reviewed.

Resource Package: This has not been addressed directly but may be addressed by the whole approach of Functional Views. It needs review to determine the purpose of Resource Package and its current use. We need to determine how these uses are supported by DDI4.

Underlying patterns: Substantial work has been done on patterns (Signification, Methodology, Collection, and Process). My perception of the Modeling Team is that these are at the point of being finetuned. We understand how they work together. We have been simplifying complex structures to facilitate implementation and are currently testing these changes. Significate work was done at the Lawrence Sprint.

Conceptual: Concepts, Universe, Population, Categories, and UnitType have been well defined and aligned with GSIM. Collection pattern work is being tested out on these classes to make sure we can manage them effectively. The Variable Cascade seems to be stable.

Geography: We have not recreated the ability to capture Geographic Structures or Geographic Locations. Recent work on Collections and StatisticalClassifications indicate that we will be incorporate these with minimal additions, primarily creating spatial relations similar to the temporal relations uses in complex processes. Once Collections are stable and Statistical Classification validated this is probably low hanging fruit in terms of getting this information into DDI4.

Capture: This includes Questions and Measures. We have the underlying structure but currently Questions are limited to QuestionItem with the peripheral contents (Instructions). Response Domains still need considerable work. There are questions about how these should relate to Value Representations for Variables and how to deal with 3.2 Response Domains that do not have Value Representation equivalents. This is priority work. In addition, work needs to be done on the support of Question Grids and Question Blocks. These were added to 3.2 and are used extensively at varying levels of complexity by statistical organizations, medical questionnaires, and educational studies.

Data Capture Flow: This is still not functional but it is anticipated that changes being made in Collection and the implications for Process pattern will solve this problem. This is a priority within the Modeling Team. We need to get this working asap.

Variables logical and physical: Initial testing of the logical description of common variables and response domains (text, date-time, numeric, code-list) has shown that the information can be captured. Some questions have been raised about the move from differentiating described value representations as separate types to having a single structure that can describe a number of value representations will be testing with the Codebook Functional View. All the specialized value representations from 3.2 have not been incorporated into DDI4.

NCubes logical and physical: It is my understanding that physical layout features of NCubes have been addressed but not all of the logical. It is unclear if this is being tested in Codebook Functional View.

Achival Information: This includes the means of creating Lifecycle Information, Access information, management of related materials, and capturing information on agents. Agents seem to be relatively stable and mature. Access information has been added but not consistently incorporated. Lifecycle information is really waiting for the Process Pattern to move past recording a design to addressing execution and the related logs of “lifecycle information”. Overall this broad set of information has been addressed in specific areas but not looked at systemically. We need a better idea of how this information is used from an information management perspective. We have begun to work on various aspects including new areas such as Data Management Plans, ingest processing, and provenance. This needs concentrated work.

# New Areas:

Process: We have a pattern and need to apply it to something other than questionnaire flow. We are in a good position to expand this in two directions, data process uses, other data capture processes that are more complex (event, parallel processing, paradata/process data capture) and from different perspectives (design, execution, reporting activities to support use of resulting data). I think we are finally in a place to start working on this effectively.

Qualitative: Currently the content entered into Drupal reflects a direct entry of external Qualitative systems (I believe from RDF). No work has been done to evaluate, test, or develop this area. It is currently not up-to-date in terms of compliance with patterns. However, some parts of the original qualitative classes have been moved out into separate packages and used by Format Description. There is some replication between classes in this section and classes brought in from DDI 3.2 resulting in multiple ways of doing things. We have been trying to reduce this type of duplication but have not done so in a structured way.

Data Capture: Changes have been incorporated to plug the use of a question to capture data into a generic data capture structure. Underlying conceptual work has been done we need to complete the incorporation of 3.2 content coverage into this conceptual base.

Data Description: Work to expand description from the level of data file and record to case and datum level has been extensive. This was needed to support new approaches to storing, discovering, access, and analyzing data. Not all aspects of 3.2 data description have been incorporated but these seem to be specialized cases. NCubes need to be more completely and clearly covered. There needs to be work done to tie this together with the rest of the model including relationships and touch points between data capture and data description. The perspective has been very much from the Datum up, which was sorely needed, but has not really been tested from the perspective of designing what data is needed, how it is captured, and how it should be represented. Minimal review of this has taken place. It has never formally come before the Modeling Team for review.