Sprint #2 (EDDI) Report

Summary

The EDDI Sprint was limited to two days. Priorities were addressing technical issues that must be resolved prior to any publication and beginning the content work on Organization/Individual, Discovery, and Process/Provenance. The selection of topics for content work was based on the expertise available during the Sprint and was intended to, at minimum, frame further discussion and provides a basic outline of higher level objects.

Deliverables

Given the short time of this Sprint, deliverables were limited:

- Finalize decisions regarding the objects used for modeling as well as modeling/design rules
- Determine the contents of a Review Package (a package of objects, such as Foundational, expressed as XML and RDF implementations, model, and accompanying documentation or other materials)
- Establish the working process between Drupal and Enterprise Architecture to capture content change content, synchronize the content of the two systems, and manage versioning
- Revise Drupal descriptions based on issues raised during and following Sprint #1
- Address XML and RDF binding issues
- Determine Automated Build Process (Production Process Flow) and document
- Review Core and Foundational packages for coverage completeness
- Provide initial coverage, review, and development plan for the following content areas:
    - Discovery
    - Organization/Individual
    - Process and Provenance

Progress on Deliverables

Prior to the Sprint a small group of people involved in the Sprint #1 content groups met to review the work done on instructions and rules for entering content into Drupal since Sprint #1. The package leaders for the Simple Instrument Group, Simple Data Set, and Foundational Group were available and plans were made to move the review and finalization of these objects forward.

- Finalize decision regarding the objects used for modeling as well as modeling/design rules
    - *In process: Allowed objects were confirmed and a draft of rules presented. The rules regarding modeling and design are being refined and finalized during the next few months. We expect there to be revisions as we work our way through the first packages.*

- Determine the contents of a Review Package (a package of objects, such as Foundational metadata, expressed as XML and RDF implementations, model, and accompanying documentation or other materials)
    - *Completed for initial packages: Review following release of initial packages*
- Working process between Drupal and Enterprise Architecture to capture content change, synchronize the contents of the two systems, and manage versioning
    - *Short term completed, Long term parked: Addition of a visualization feature to the Drupal content and means of managing versions of the XSI output from Drupal has provided a short-term solution for capturing change within Drupal. A round-trip solution supporting capturing change within EA and porting it back to Drupal has been parked. It is still the preferred solution, but will require more time to accomplish.*
- Revise Drupal based on issues raised during and following Sprint #1
    - *In process (minor revisions still being made based on new requirements): Visualization, process steps, 3.2 content, and reporting features have been added to Drupal as identified.*
- Address XML and RDF binding issues
    - *In process: XML has been completed except for definition of required extended primitives: All XSLT's are under version control.*
    - *In process: RDF document has been reviewed. There are still some outstanding governance issues as well as an issue dealing with ordering when needed. Production techniques still require testing to identify the optimal solution that results in consistent RDF structures.*
- Determine Automated Build Process (Production Process Flow) and document
    - *In process: Build steps and process requirements have been identified*
- Review Core and Foundational packages for coverage completeness
    - *In process: Reviewed Core objects and revised. These will be compared to a list of objects used in 3.2 as Types and Extension bases post-sprint.*
    - *In process: Foundational content will be reviewed post-sprint within content group.*
- Provide initial coverage, review, and development plan for the following content areas:
    - Discovery
        - *In process: Content coverage completed, required objects identified; determined that this is primarily a view of DDI. Content entered in Drupal - Stage: In Progress.*
    - Organization/Individual
        - *In process: Content coverage completed, carry-over from 3.2 structures decided, and objects Agent and SoftwareAgent entered in Drupal - Stage: In Progress. Other objects will be based on 3.2 structures loaded into Drupal post-sprint.*
    - Process and Provenance
        - *In process: Content coverage completed. Identified models needed for requirement reviews. Plans for continued work post-sprint completed.*

Meeting Notes

Technical Issues

XML Binding document

- Have completed all but definition of which extended primitives are needed
- All the XSLT's are under version control

RDF Binding

- Governance issues outstanding
- Ordering – when it's required by the model – need to be able to identify where ordering is important and what is the construct used for this in the model
- Reviewed full document
- There is some literature on the web on this which should be reviewed regarding how to do the transformation
- Need to test that you get the same result from different production techniques (need to be consistent and make the right choice now)

Review package

- Determined the content of the review package in terms of documentation
- Identified some changes in Drupal to support
- Interaction between XMI coming back into Drupal regarding information flow and source of versions
- Relationships – are the three categories enough and confusion around "Neither"
- Discussed process flow – started but needs to continue
- Binding -- generation of documentation which is implementation-specific
- How should the implementation of specific documentation happen? (suggested at Rule level rather than Object level)
    - The discussion at Dagstuhl was to keep these separate things in a binding configuration
    - Where should it be formalized? UML? Binding document? Off in its own package?

Automated Build Tasks (Production Process Flow)

Build Steps

Jenkins or Cruise Control can be installed on the Lion server. Upon check-in to SVN or change to Drupal content, automatically perform the following steps:

1. Run the XMI generator
2. Diagrams are automatically created in the browser
3. Run XSLT transform to generate XML Schema
4. Run XSLT to generate RDF

5. Run XSLT from XMI and Docbook to generate documentation
6. Create Field-level documentation from XSD
7. Tests
    a. Validate the XMI
    b. Validate XSD
    c. Validate RDF?
8. Reports
    a. Test results
    b. Burndown: How much of DDI 3.2 is covered?
    c. Email notifications
    d. Web reports

Requirements

- Automatically check cross-dependencies within a package
- A tool easy for content people to use
- A tool in which modelers can make corrections and changes easily, too
- Automatically keep the model and documentation in sync
- Diagrams as part of the documentation:
    - Object level and relationship to neighbors
    - Package level
    - Put these in the Docbook
    - Printable (PDF)
- Generated XMI will be consumable by Enterprise Architect, and ideally other UML tools as well
- Track who is editor of each object and package level
- Generate one XMI file per package
- Generation should occur on a fixed interval (daily?)
- Check generated XMI files into version control
- Version number of package. Could use the SVN revision number

Requirements additions:

- Graphics workable in IE 9 on and common versions of Firefox
- Find out which XMI fields to store the documentation in for transformation into XSD and RDF

*Fast and Automated Semantic Transforms (FAST)

PARKING LOT

- Round tripping between Drupal and EA
- Identification, Versionable, Maintainable (Label, description, etc.)
- Identification of base types
- TOFKAS – decide correct name for this, which stands for “Object Formerly Known As Study”

- “Reference” and naming rules

Content Topics

All content topics have at minimum a working definition of what they are covering and a process for continuing this work. All will need broader review and may be carry-over topics for Sprint #3 (NADDI).

Discovery

Objects used for Discovery

- **Coverage (topical, temporal, spatial)
- “All text fields” (code value, name, label, description)
- Study
- Data file
- Questions
- Concepts
- Universe
- Data
- A level between the information model and an application model
- **Citation information (title, creator, producer, publication/production date, plus hasPart, isVersionOf)
- Faceted search

Starting list of topics:

- Title -> Citation
- Author -> Citation
- Abstract
- Link to object (Related publications, objects of study)
- Universe
- Kind of data
- Topical Coverage (keywords, etc.)*
- Geographical Coverage*
- Temporal Coverage*
- Generalized model?

There is an object “Coverage”: this is made up of spatial, topical, and temporal converges. Each of these is a super-class which might have different sub-types.

For Spatial Coverage, we identified at least three sub-types: Bounding Polygon (which might need to be split into two: bounding boxes and full geographical structures/polygons); Coded Geographies; and External GIS systems (pointer to system). It might be nothing more than a label (the word “Paris”).

For Topical coverage, we have a similar situation: Topical Coverage is a super-class with multiple sub-types: Subject, Keyword, and External system (like a folksonomy). Subjects and Keywords are identical in structure and description in 3.2: are they really the same thing?

There is the idea that an abstract Category Structure or "Category System" could be devised, from which subtypes of formal subject matter classifications and thesauri could be specialized. Categorization objects would connect identifiable things to a node in the category structure. The values of subject and category would be specializations of the Categorization object, to which restrictions might be attached (only for Variables and Studies, etc.) Concept tie-backs are gotten for free through the Categories, which are Concepts used in the role of Categories.

Temporal Coverage – We need better definitions here: is this a reference period, or something else? Could the structure of this date be described as a primitive from the perspective of the model (structure would be in the binding rules)?

What is the relationship between Universe and Coverage? Universe was originally only for describing the locale of a respondent within the administered flow of a questionnaire. It was not originally intended to be a full-blown description of a population. It is not the same thing when applied to the questionnaire. Look at GSIM 1.1 to review relationship between unit, unit type, and population.

Coverage fits into one of the filters into the intended set of what you are trying to study. This gets into issues of sampling methodology, etc. There are always restrictions on your population ("filters").

Jay says: They use population ("Universe") plus Coverage – there is more information here than simply the Coverage information. Example: NCS: Interviewing the mother about the child and the mother at the same time.

Link to object:

Two cases were considered: the publication of research based on data, and the links to external objects used in data collection (e.g., stimuli such as images, etc.).

For the first case, it was felt that links from published research should be from the research to the data (via the citation provided in the metadata), thus off-loading the burden of this type of discovery onto citation databases and discovery systems. Thus – nothing is needed in the DDI itself.

In the second case, links here are already in the DDI description of surveys, and presumably this could be extended for other types of instruments. For discovery purposes, this metadata could be "mined" from the detailed descriptions for supporting discovery systems. Thus – nothing needed in this area of DDI itself.

Organization / Individual

After reviewing the Process model in GSIM as well as the DDI 3.2 structure, it was determined that the current 3.2 structure will be adopted with the following changes:

1. Creation of an abstract class “Agent” as an extension base for other types of Agents. Note that any packaging structure should reflect the idea of an Agent as opposed to the current “Organization Scheme”.
2. For both Organization Agent and Individual Agent collapse the “FormalXxxxName” into the XxxxName within XxxxIdentification adding a Boolean attribute isFormalName.
3. Add a SoftwareAgent modeled on DDI 3.2 Software structure.
4. Packaging structure for Agents needs to take into account the current structure “Relation” which links two Agents and describes the role of the target to the source.
5. Jon Johnson will enter these objects into Drupal

Process / Provenance

The group reviewed the GSIM Process structure and the W3C Prov model for Provenance. Some examination was made prior to the group discussion regarding the relationship of Prov to the PREMIS structure heavily used within archives. Prov has mappings to PREMIS and the Relation structure in Organization / Individual supports the attachment of Access Rights to individuals and organizations. It was felt that the availability of this structure addressed the limitations of Prov in the archival context.

This group will continue to work following Sprint #2 and the content can be reviewed at Sprint #3 (NADDI). Denis Grofils is the group leader and will develop a plan over the next few weeks which will result in a model that can be entered in Drupal. The need for a generic model was noted. The model in GSIM is advertised as a “statistical process model”. This needs to be well tested with the range of process model needs at various points within the DDI coverage area, including the case of “software as a service” and provenance chaining. The DDI model must be abstract enough to support future development and applications.

The scope of DDI 4.0 process metadata must include:

- Business process description
- Data processing description
- Business process execution
- Business monitoring activity

Existing standards will be reviewed during this process.

Some discussion areas (notes)

- Description level
- Execution level
- Process definition at design time vs. execution time
- Data/metadata as service
- Provenance chaining, i.e., microdata -> aggregate data
- Possible requirement:
    - Mapping of GSIM process model to products (business process management), i.e., SAS BI, Talend (open source)

- BPMN very complex, how to constrain? Can it be usable?
- BPEL is on the implementation level
- Is BPEL in parallel to DDI, SMDX on the implementation level below GSIM or should there be a stronger relationship between DDI and BPEL?
- Is there a binding possible?