

# Structured Re-usable Documentation for DDI 4

Prepared by: Oliver Hopt, Jon Johnson, Amber Leahey

## Introduction & Rationale

The documentation of DDI can be used for different purposes and for different audiences. The DDI 4 model is currently structured in such a way that makes the re-use of documentation difficult. A key goal of the working group is to identify structures and workflows that are better optimized for documentation re-use.

Changes to the documentation process could be made using such a structured approach. As new classes in the model are finalized, it would be required that this structure be completely populated to support the creation and re-use of documentation at the class-level. A class will thus be accompanied by rich documentation as well as the standard property and relationship sets.

The new documentation structure aims to meet two requirements:

- 1) to support re-use of documentation for different outputs based on the model;
- 2) enabling changes to the modelling process to capture better documentation

At a high-level, documentation flows through the modelling workflow for the production of DDI. As documentation becomes more structured with a view to certain purposes and audiences, it can be re-used throughout the model. At the class-level, the capture of additional types of documentation to support re-use greatly enhances the modelling process, reduces inefficiencies, increases understanding, and, enables the re-use of documentation for different purposes and audiences.

Audiences are particularly important for defining how documentation can be re-used, as certain audiences require certain types of documentation. Consideration of the audience, supports a view to reduce documentation burden, improve workflows across the model, and establish greater understanding and support for documentation re-use and training.

Finally a set of identified structured documentation approaches and guidelines are provided at all levels (high-level, functional view-level, and class-level) that can be applied to the model production, which supports the creation of rich documentation and re-use.

## Section 1: Establishing a structured reusable documentation approach

In general, documentation should ultimately aim to improve people's understanding of DDI and how to use it. Structured documentation is required for the systematic re-use of information about how to document data in the DDI 4 model. Researchers and data documenters should be able to use a particular DDI View and document their data without knowing or understanding the underlying model for the standard. In order to adopt this kind of approach to documentation, we need to identify the important elements of documentation

creation, the audiences, considerations and constraints that may determine use of documentation, and ideally how this will be used or made visible to users.

**An overview approach for establishing structured re-usable documentation in DDI 4:**

1. Identify audiences
2. Things to describe (model, classes, views)
3. How to do the things (technical people, non-technical people)
4. How to surface the things to different audiences

Using a structured documentation approach, it's possible to address the documentation needs of some identified set of audiences, such as ourselves (DDI Moving Forward), deciders (managers, administrators), implementers (developers), and data documenters (data managers, researchers, etc.). Documentation requirements for the different audiences, includes identifying high-level and class-level documentation required for supporting migration by current users of DDI (Codebook and Lifecycle), linkages to glossaries, vocabularies, and definitions, and the requirements for defining relationships to existing documentation.

The following provides an overview of the documentation structure at the class-level and functional-view level. Moving forward all class items would be populated with the required documentation to support the goal of structured and re-usable documentation across the model. At the functional view level, documentation will take granular class items and provide users with what is recommended for use based on that functional-view.

**Structured documentation at the Class-Level (CL) and Functional-View Level (FL):**

<b>Structured Documentation</b>	<b>Class-Level (CL)</b>	<b>Functional-Level (FL)</b>
Formal Definition	<b>Y</b>	<b>Y</b>
Purpose	<b>Y</b>	<b>Y</b>
Relationship to existing documentation	<b>Y</b>	<b>Y (Workflows)</b>
Other specifications / domains	<b>Y</b>	
Links to existing glossaries, vocabularies, definitions, etc.	<b>Y</b>	
Technical examples	<b>Y</b>	<b>Y (Use cases)</b>

Implementing a structured documentation approach considers the production workflows that can support documentation re-use. Therefore to ensure that documentation is improved and

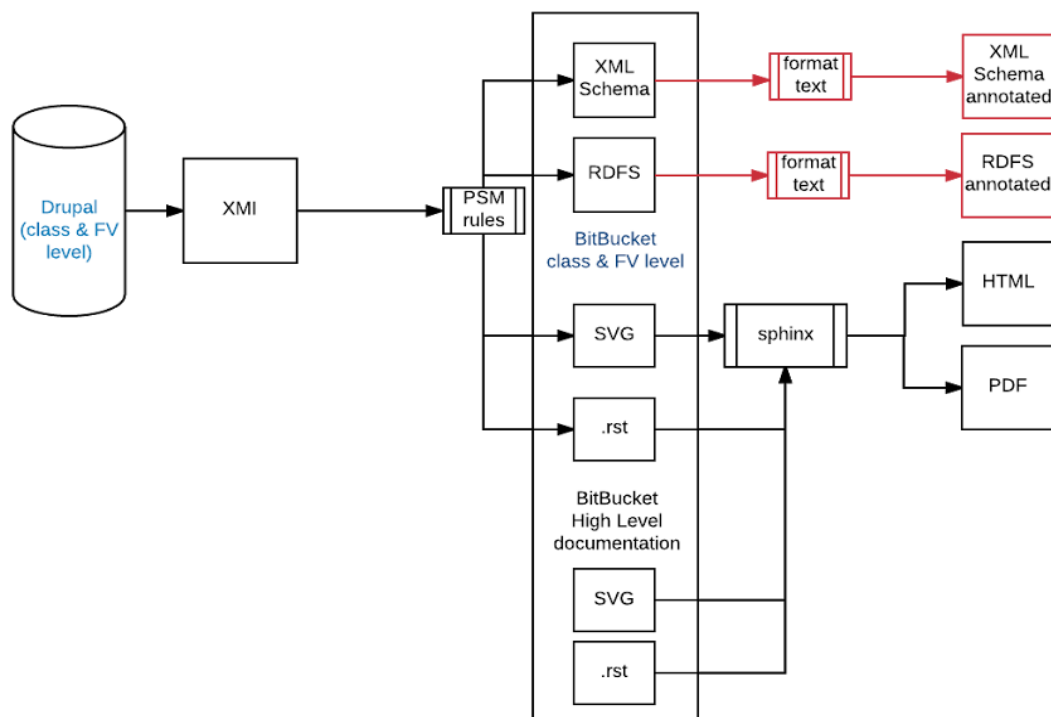
structured, as documentation is captured at the varying levels, an evaluation of the different documentation production workflows is necessary.

## Section 2: Structured re-usable documentation workflows

Drupal, the documentation tool used for the creation of the DDI 4 model, supports structured and versioned documentation for classes, views, and packages. Using the high-level overview of the model production workflow, and thinking about re-usable documentation, the goal is to produce as much as possible *loss/less* class-level documentation and incorporate structured documentation in the production of the DDI outputs (XML, RDF, etc.). The export and inclusion of all class-level elements into the XMI from Drupal ensures richer documentation of the model and uses of it.

### DDI 4 Documentation Production Model Workflow

Figure 1 - High-Level (HL)



*The above diagram represents the documentation model production workflow (the transfer, exchange, and reuse of documentation in the DDI 4 model).*

In order to facilitate the production and re-use of structured documentation for different purposes and different audiences, we currently need a mechanism for documenting recommended and excluded class-level documentation to be re-used in the creation of functional views and audience specific documentation.

## Recommended documentation inclusions to support reuse

Figure 2 - Documentation inclusions at the Class-Level (CL)

**In green** – include under Definition field in the XMI (model)

**M – Mandatory**

**T - Technical**

**TL - Technical Lite**

Field	Output	XMI fields	Type
Name	Y	Y	M
Package	Y	Y	T
Extends	Y	Y	T
Is Extendable	N	N	
Is Property	N	N	
Status	N	N	
Stage	N	N	
Version*	Incorporate Drupal versioning - Y	?	M
<b>DDI 3.2</b>	Y	N	TL
<b>GSIM</b>	Y	N	TL
<b>RDF Mapping</b>	Y	N	TL
<b>Contributor</b>	N	N	
<b>Definition</b>	Y	Y	M
<b>Example</b>	Y	N	M
<b>Synonyms</b>	Y	N	M
<b>Explanatory Notes</b>	Y	N	M

Overage Image	N		
Properties*	Y	Y	M
Relationships*	Y	Y	M
PackageDescriptionName	Y	Y	M
FunctionalView	Y	Y	M

-add list of additional items to include & why they are needed

Controlled vocabularies

### Section 3: Documentation for different audiences

Documentation can be structured and made re-usable if the incorporation of documentation “type” tags are added to class items. This can ensure certain types of documentation are provided to specific audiences. The following identifies a set of audiences and their documentation needs.

Figure 3 - Audience documentation needs / types

Audience	Types
Ourselves (Moving Forward)	everything
Deciders	M – High level
Developers	M + Technical
Documenters	M + T Lite
Researchers	M

Explain audience documentation needs (add from today)

Researchers & data documenters

Section 4: Examples and guidelines for use

Workflow Step example

# Appendix: List of issues reported to JIRA (Structured re-usable documentation WG)

JIRA tasks, improvements, and new features added

	A	B	C	D	E
1	Summary	Issue key	Issue id	Issue Type	Status
2	Add functionality to process EA Definition XMI field for .rst transform	DOC-14	19719	Improvement	Open
3	Improve communication about collections and type of collections	DOC-20	19754	Improvement	Open
4	Import XMI into alternative tool	DOC-13	19718	New Feature	Open
5	Clickable dependencies on Drupal Diagrams	DOC-12	19717	Improvement	Open
6	Diagrams for views in drupal	DOC-9	19714	Improvement	Open
7	Contributor is difficult to select	DOC-7	19712	Improvement	Open
8	extra properties to be exported into XMI	DOC-4	19703	New Feature	Open
9	Preparation for Dagstuhl Sprint 2016	DOC-1	19402	Task	To Do
10	Add field to drupal for pattern associations	DOC-19	19753	Task	Open
11	Add item entry options for DDI 3.2 field documentation to support mapping	DOC-18	19752	Improvement	Open
12	Add a mechanism for qualifying equivalencies between specifications and versions (DDI 3.2, RDF Mapping etc.)	DOC-17	19751	Improvement	Open
13	Develop training for Durpal 'best practices'	DOC-16	19750	Task	Open
14	Add a mechanism for changes that cause pattern disruptions to be flagged for review	DOC-15	19749	Improvement	Open
15	Colouring of associations and inheritance	DOC-11	19716	Improvement	Open
16	DDI3.2 Mapping - make mandatory	DOC-10	19715	Improvement	Open
17	New field in Drupal to reference usage of ExternalControlled Vocabularies	DOC-5	19707	Improvement	Open
18	incorporate ordering of properties and relationships	DOC-8	19713	Bug	Open
19	Fields not present in view mode if they are null	DOC-6	19711	Improvement	Open
20	Adapted documentation for different audiences	DOC-3	19604	New Feature	Open
21	How to structure the descriptive text of high level elements like packages and views	DOC-2	19603	New Feature	Open

## Progress to date

Project: **Documentation Work**

Chart

This chart shows the number of issues **created** vs. the number of issues **resolved** in the last 30 days.

