Course on DDI 3: Putting DDI to Work for You

3 December 2009
Wendy Thomas, MPC

1st Annual European DDI Users Group Meeting:
DDI – The Basis of Managing the Data Life Cycle

Credits

- The slides were developed for several DDI workshops at IASSIST conferences, various special presentations, and at GESIS training in Dagstuhl/Germany
- Major contributors are Wendy Thomas and Arofan Gregory
- Further contributors are Joachim Wackerow and Pascal Heus

Introductions

- **About yourself:**
  - Who you are?
  - What organization you're with?
  - Do you consider yourself a technical person in terms of IT?
- **About your work:**
  - What do you do?
  - Will you have to train other people or supervise their work related to DDI?
  - What other metadata standards have you worked with?
- **Why are you interested in DDI?**
- **What are you here to learn?**

Outline

- Why DDI content was expanded and the goals of DDI 3 and beyond
- What applications/use cases DDI 3 is designed to support and what earlier versions can support
- How DDI relates to other standards (where does it fit into the standards picture)
- Future development directions

Background

- Concept of DDI and definition of needs grew out of the data archival community
- Established in 1995 as a grant funded project initiated and organized by ICPSR
- Members:
  - Social Science Data Archives (US, Canada, Europe)
  - Statistical data producers (including US Bureau of the Census, the US Bureau of Labor Statistics, Statistics Canada and Health Canada)
- February 2003 – Formation of DDI Alliance
  - Membership based alliance
  - Formalized development procedures

Characteristics of DDI 1.0/2.0

- Focuses on the static object of a codebook
- Designed for limited uses
  - End user data discovery via the variable or high level study identification (bibliographic)
  - Only heavily structured content relates to information used to drive statistical analysis
- Coverage is focused on single study, single data file, simple survey and aggregate data files
- Variable contains majority of information (question, categories, data typing, physical storage information, statistics)

Impact of these limitations

- Treated as an “add on” to the data collection process
- Focus is on the data end product and end users (static)
- Limited tools for creation or exploitation
- The Variable must exist before metadata can be created
- Producers hesitant to take up DDI creation because it is a cost and does not support their development or collection process

DDI 1/2.x Tools

- Nesstar
  - Nesstar Publisher, Nesstar Server
- International Household Survey Network (IHSN)
  - Microdata Management Toolkit
  - NADA (online catalog for national data archives)
  - Archivist / Reviewer Guidelines
- Other tools
  - UKDA DExT
  - ODaF DeXtris
  - http://tools.ddialliance.org

Why the major change?

- DDI 3 is a major change from DDI 2 in terms of content and structure. Lets step back and look at:
  - Basic differences between DDI 2 and DDI 3
  - Differences in functionality, what is DDI 3 good for
  - Review a few general application possibilities

Differences Between DDI 2 and 3

- DDI 2
  - Codebook based
  - Format XML DTD
  - After-the-fact
  - Static
  - Metadata replicated
  - Simple study
  - Limited physical storage options
- DDI 3
  - Lifecycle based
  - Format XML Schema
  - Point of occurrence
  - Dynamic
  - Metadata reused
  - Simple study, series, grouping, inter-study comparison
  - Unlimited physical storage options

DDI What Is It Good For?

- There are some obvious differences between DDI 2.* and DDI 3.*
  - Ability to capture comparative information
  - Ability to re-use and share metadata
  - Ability to mark up data in XML
  - Greater ability to facilitate data discovery and relationships
  - It is designed to capture lifecycle information as it occurs, and in a way that is useful during production
  - It is machine-actionable – not just documentary
- All of this comes with added complexity
- It also allows for greater interoperable support between organizations
- Here are a few examples…

Upstream Metadata Capture

- Because there is support throughout the lifecycle, you can capture the metadata as it occurs
- It is re-useable throughout the lifecycle
  - It is versionable as it is modified across the lifecycle
- It supports production at each stage of the lifecycle
  - It moves into and out of the software tools used at each stage

Reuse of Metadata

- You can reuse many types of metadata, benefitting from the work of others
  - Concepts
  - Variables
  - Categories and codes
  - Geography
  - Questions
- Promotes interoperability and standardization across organizations
- Can capture (and re-use) common cross-walks

Virtual Data

- When researchers use data, they often combine variables from several sources
  - This can be viewed as a “virtual” data set
  - The re-coding and processing can be captured as useful metadata
  - The researcher’s data set can be re-created from this metadata
  - Comparability of data from several sources can be expressed

Mining the Archive

- With metadata about relationships and structural similarities
  - You can automatically identify potentially comparable data sets
  - You can navigate the archive's contents at a high level
  - You have much better detail at a low level across divergent data sets

Overview of DDI 3

DDI 3.0 and 3.1

- DDI 3.1 was published 2009-10-18
- REMEMBER when we refer to DDI 3 we mean both 3.0 and 3.1
- When we talk about DDI 3.1 the feature we are talking about is not available in DDI 3.0
- We are teaching the latest version which is DDI 3.1

Origins of the DDI Alliance

- Versions 1.* and 2.* were developed by an informal network of individuals from the social science community and official statistics
  - Funding was through grants
- It was decided that a more formal organization would help to drive the development of the standard forward
  - Many new features were requested
  - The DDI Alliance was born to facilitate the development in a consistent and on-going fashion

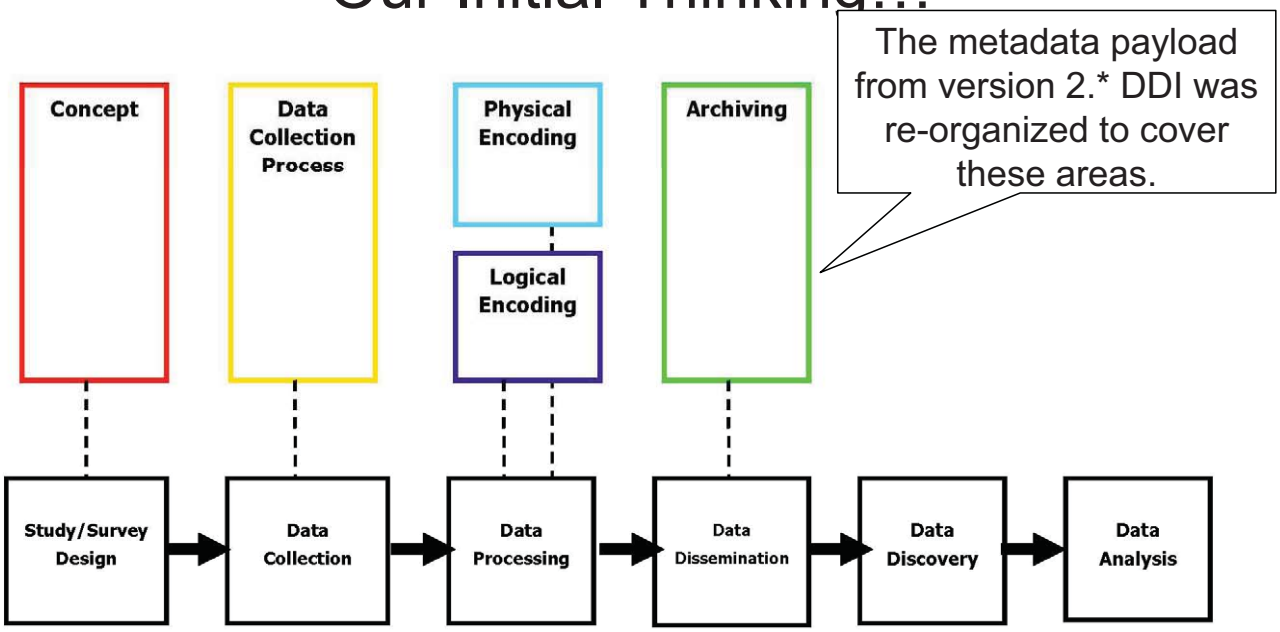
DDI Alliance Structure

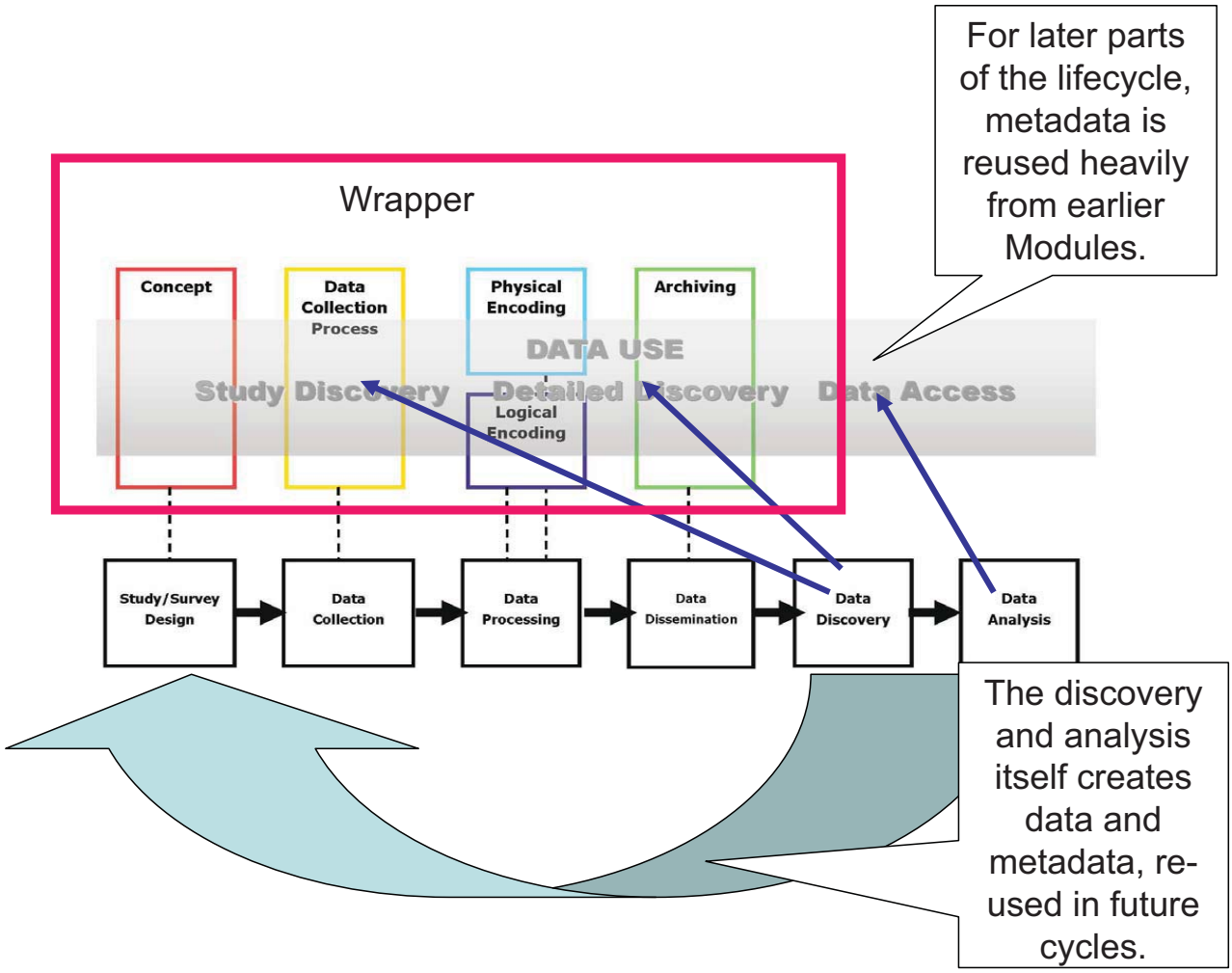
- DDI 3.0 (and future) specifications were created by committees drawn from among the member organizations
  - Some outside experts are invited to attend
- The Steering Committee governs the organization
- The Expert Committee votes to approve all published work
  - One representative per member organization
- The Technical Implementation Committee (TIC - formerly the "SRG") creates the technical work products (XML schemas, UML models, documentation, etc.)
- Each subject area has a working committee to determine the needed metadata (for example, Survey Design & Instrumentation)
- Usability and Outreach Group promotes the standards and performs liaison with other standards groups

Requirements for 3.0

- Improve and expand the machine-actionable aspects of the DDI to support programming and software systems
- Support CAI instruments through expanded description of the questionnaire (content and question flow)
- Support the description of data series (longitudinal surveys, panel studies, recurring waves, etc.)
- Support comparison, in particular comparison by design but also comparison-after-the fact (harmonization)
- Improve support for describing complex data files (record and file linkages)
- Provide improved support for geographic content to facilitate linking to geographic files (shape files, boundary files, etc.)

Our Initial Thinking…





Realizations

- Many different organizations and individuals are involved throughout this process
  - This places an emphasis on versioning and exchange between different systems
- There is potentially a huge amount of metadata reuse throughout an iterative cycle
  - We needed to make the metadata as reusable as possible
- Every organization acts as an “archive” (that is, a maintainer and disseminator) at some point in the lifecycle
  - When we say “archive” in DDI 3, it refers to this function

DDI 3 Lifecycle Model

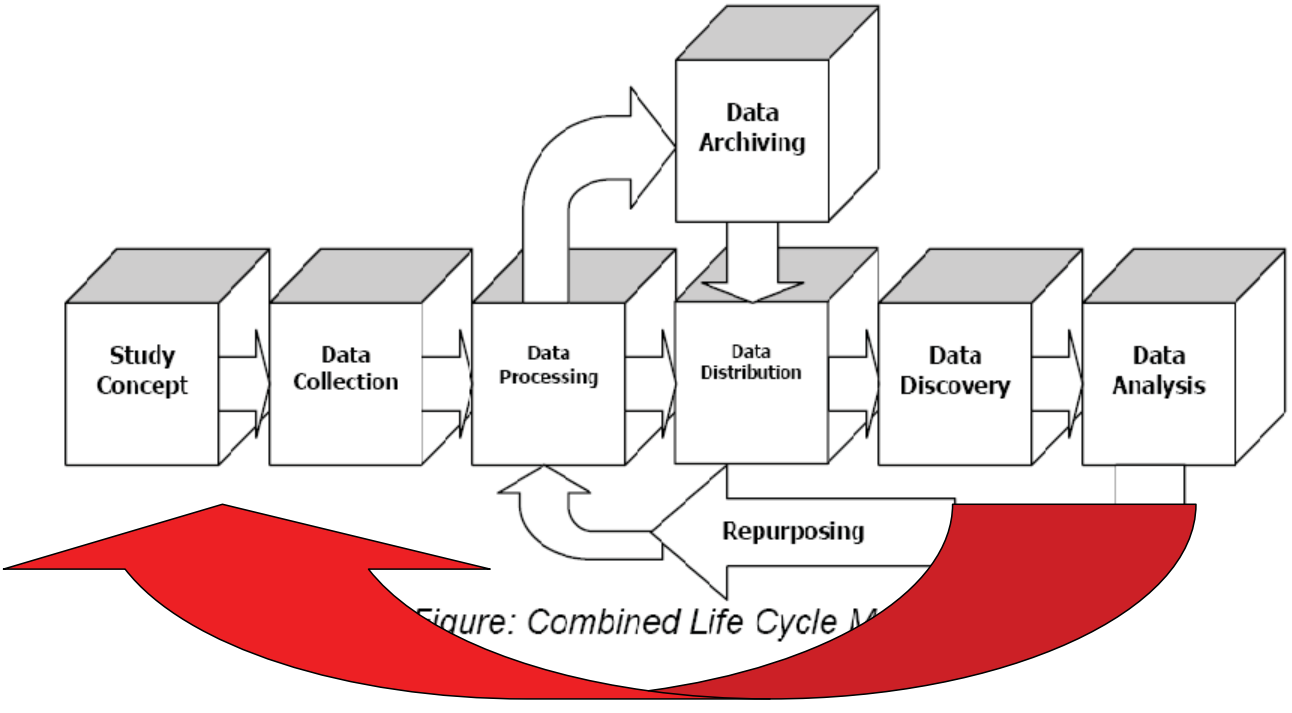
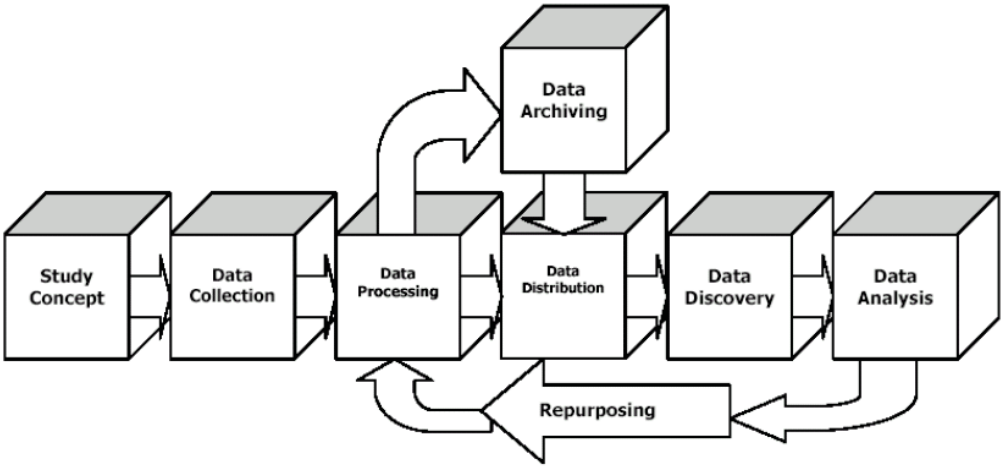


Figure: Combined Life Cycle M

Metadata Reuse

DDI 3 and the Data Life Cycle



- A survey is not a static process: It dynamically evolved across time and involves many agencies/individuals
- DDI 2.x is about archiving, DDI 3 across the entire “life cycle”
- DDI 3 focuses on metadata reuse (minimizes redundancies/discrepancies, support comparison)
- Also supports multilingual, grouping, geography, and others
- DDI 3 is extensible

Life Cycle Orientation

DDI 3.0 documents all stages in the life cycle of a data collection:

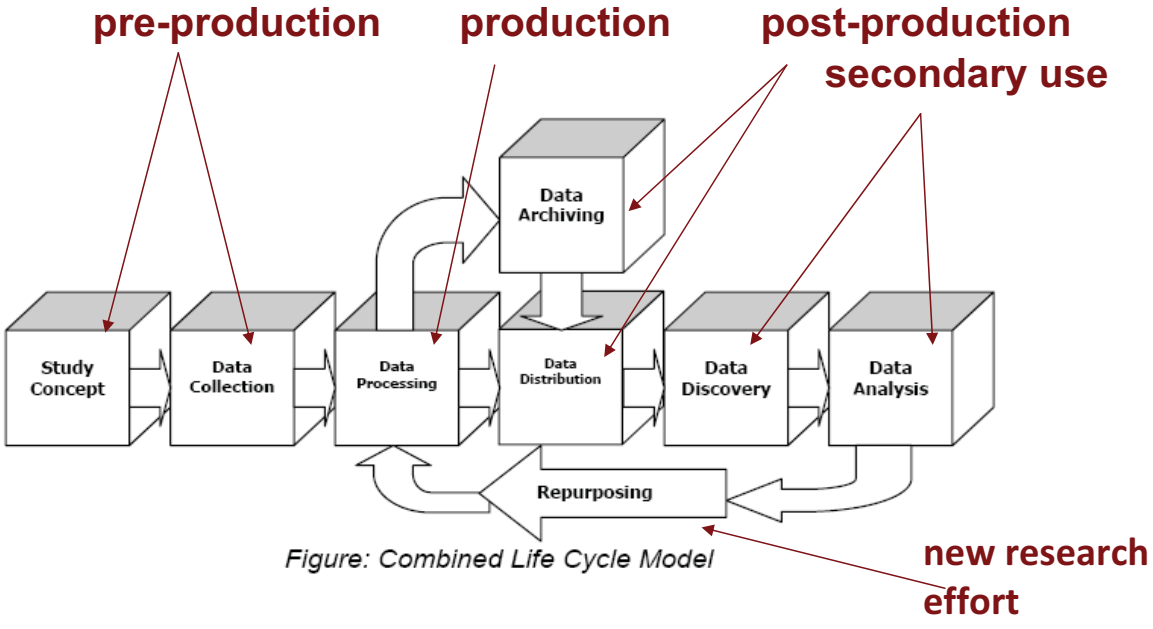


Figure: Combined Life Cycle Model

Approach

- Shift from the codebook centric model of early versions of DDI to a lifecycle model, providing metadata support from data study conception through analysis and repurposing of data
- Shift from an XML Data Type Definition (DTD) to an XML Schema model to support the lifecycle model, reuse of content and increased controls to support programming needs
- Redefine a “single DDI instance” to include a “simple instance” similar to DDI 1/2 which covered a single study and “complex instances” covering groups of related studies. Allow a single study description to contain multiple data products (for example, a microdata file and aggregate products created from the same data collection).
- Incorporate the requested functionality in the first published edition

Development of DDI 3.0

- 2004 – Acceptance of a new DDI paradigm
  - Lifecycle model
  - Shift from the codebook centric / variable centric model to capturing the lifecycle of data
  - Agreement on expanded areas of coverage
- 2005
  - Presentation of schema structure
  - Focus on points of metadata creation and reuse
- 2006
  - Presentation of first complete 3.0 model
  - Internal and public review
- 2007
  - Vote to move to Candidate Version
  - Establishment of a set of use cases to test application and implementation
- 2008
  - April: DDI 3.0 published

DDI 3.1

- Contains corrections for bugs and feature corrections identified during the first year of use
- Approved for publication in May 2009
- Published October 2009
- Some changes are backward incompatible
- Structure of the DDI version number: X.Y.Z where:
  - X major new features or changes
  - Y minor incompatible changes
  - Z minor compatible changes

Continued Development

- DDI Alliance was formed to provide on-going support for DDI development
- Member driven / User driven
- Development within the parameters of related standards to exploit strong areas of information overlap and transference of metadata from one standards “world” to another

Intent of DDI Design

- Facilitate point-of-origin capture of metadata
- Reuse of metadata to support:
  - Consistency and accuracy of metadata content
  - Provide internal and external implicit comparisons
  - Support external registries of concepts, questions, variables, etc.
  - Metadata driven processing
- Provide clear paths of interaction with other major standards

Basic Structures

- DDI 3 used a model similar to SDMX in terms of the following:
  - Indentifiable, Versionable, and Maintainable objects
  - The use of multiple schemas to describe different process sub-sections in the life-cycle
  - Use of schemes to facilitate reuse of common materials

DDI: Full content coverage for survey and administrative data

- Conceptual coverage
- Methodology
- Data Collection
- Processing – cleaning, paradata
- Recoding and derivations
- Variable and tabular content
- Internal relationships
- Physical storage
- Data management

Plus: Relationships between studies

- Comparison by design
  - Study series can inherit from earlier metadata
  - Capture changes only
- Data integration
  - Mapping of codes between source and target
  - Capture comparison information
- Comparison of abstract content models
  - Publication of reusable materials (code schemes, concept schemes, geographic structure, etc.)

Current Areas of DDI Development

- Controlled vocabularies to improve machine actionability
- Data collection methodology and process expansion for more depth and detail
- Qualitative data
- Increased comparison coverage
- Tools

Reuse

- DDI is designed around schemes (lists of items) for commonly reused information within a study such as categories, code schemes, concepts, universe, etc.
  - Items are “used” in multiple locations in a DDI document by referencing the item in the list
  - Enter once, use in multiple locations
  - Items can be versioned for management over time without having to change content in multiple locations

Life Cycle
Capturing and Reusing Metadata

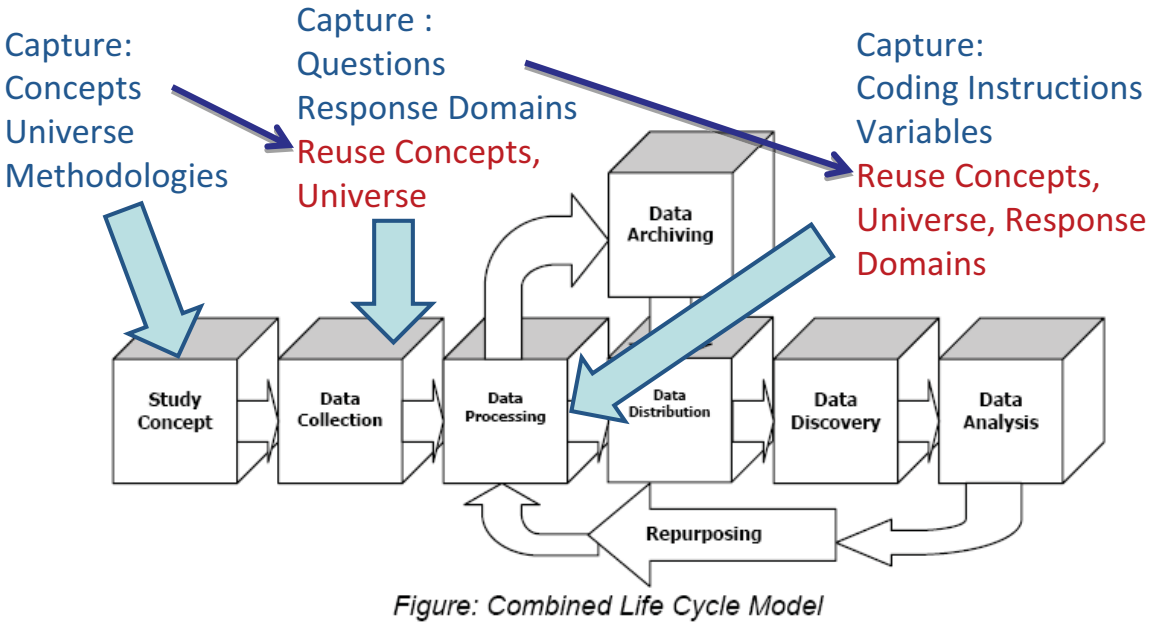


Figure: Combined Life Cycle Model

Reuse of metadata within a DDI Instance provides explicit comparability

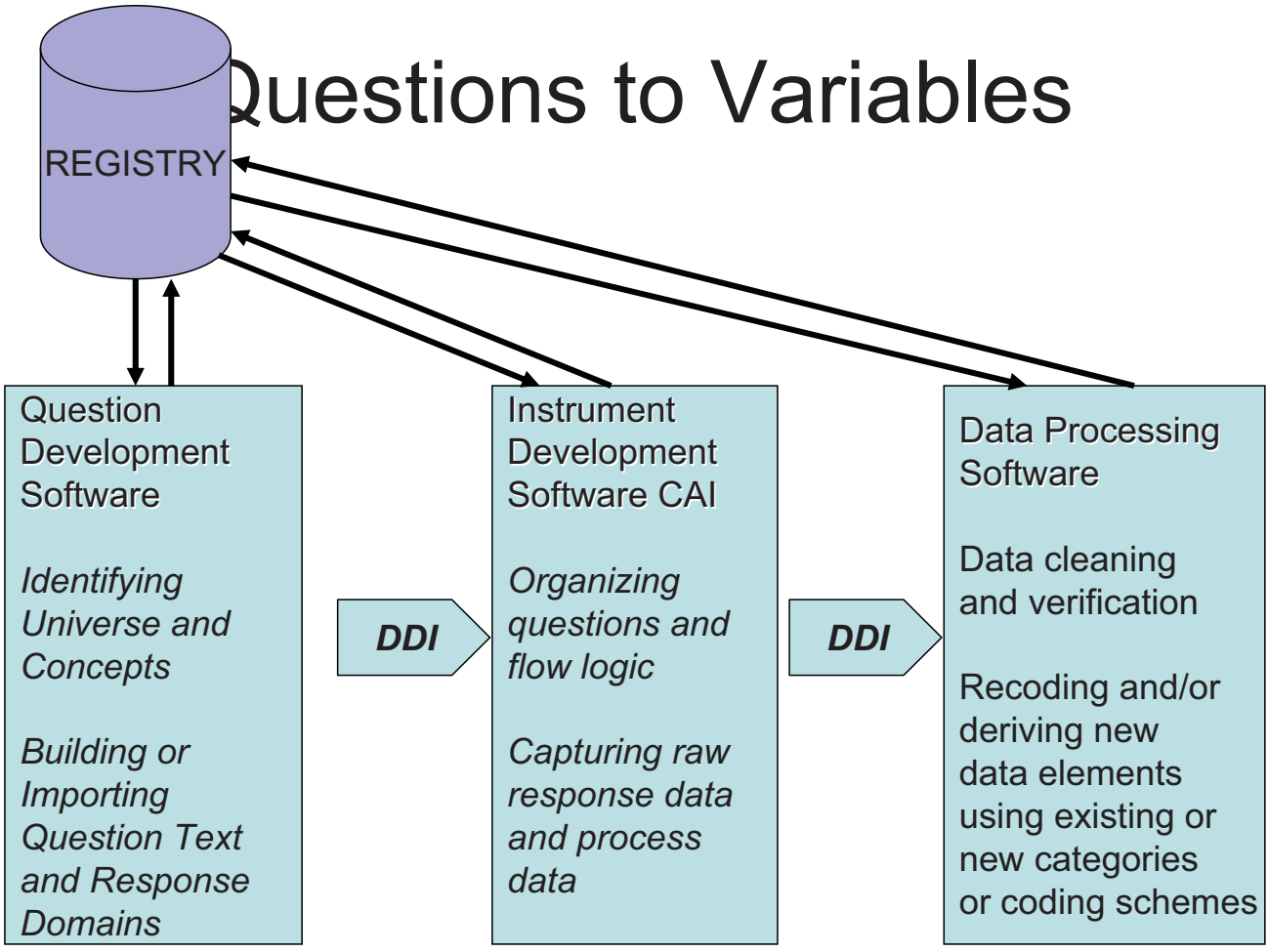
Comparison and Registries

- Information in DDI schemes can be published in external registries and used by multiple studies
  - Provides implicit comparison both within a study and between studies
  - Supports organizational consistency through the use of agreed content managed in registries
  - Referencing structured lists provides further context to individual items used in a study

Metadata driven processing

- Capturing metadata upstream can provide over 90% of the building blocks needed to generate the remainder of the metadata
- DDI supports imbedding command code to run data processing events driving data capture, data processing during after collection, and to support post-collection recoding, derivations, and harmonization maps

Questions to Variables



Working with other standards

- There is no single standard that does it all
- DDI was specifically designed to support easy interaction with:
  - Dublin Core – mapping of citation elements and imbedding native Dublin Core
  - ISO/IEC 11179 – working with an editor of the standard to reflect data element model and ISO/IEC 11179-5 naming conventions for registry intended items

Standards continued

- SDMX – DDI NCubes were revised to incorporate the ability to attach attributes to any area of a cube and map cleanly into and out of SDMX cubes. SDMX has added means of attaching fragments of DDI which provide source and processing information that can be indexed and delivered through SMDX tools.
- ISO 19115 (ANZLIC, FGDC) – Geographic elements in DDI are structured to reflect basic discovery elements used by geographic search engines and provide the detailed geographic structure information needed by GIS system to incorporate the data accurately

Value

- Supports consistent use concepts, questions, variables, etc. throughout organization
- Supports implicit comparison through reuse of content
- Supports explicit comparison by mapping content between studies and to standard content
- Retention of explicit relationships between data collection and the resulting data files
- Early capture of a broad range of metadata at point of creation

Value - continued

- Interoperability
- Flexibility in data storage
- Reuse of element structures
- Strong data typing
- Improved data mining between and across systems
- Improved access to detailed metadata

DDI Overall Structure and Component Parts

Features to Support Functionality

- Machine actionability
  - Tighter data type definitions
  - Increased use of controlled vocabularies
- Reuse
  - Shift from DTD to XML Schema
  - DDI Schemes (shared metadata)
  - Identification, Versioning, Reference
- Integration with Non-DDI materials
  - Other Material
  - Clear mapping or direct use of common element sets

Data Typing in Schemas

- More control in terms of defining legal content
  - imposed a regular expression to define the legal structure of a version or a DDI urn
- Multiple elements can reference the same element structure
  - Abstract, Purpose and other similar elements all reference the element IdentifiableStructuredString

Controlled Vocabularies

- Limited the number of internal controlled vocabularies to those that were comprehensive and relatively static
- Allowed for use of a controlled vocabulary in a wide variety of places using an established standard to publish controlled vocabularies outside of DDI
- Use Genericode with a DDI defined profile which allows for secondary validation of controlled vocabulary content

Reuse

- DTDs are nested structures
  - Content is typically replicated when used multiple times (i.e., Concept within Variable)
  - Top level of nested items need to exist before entering content of sub-elements (i.e., Variable needed to exist before entering question content)
- XML Schemas are like relational databases
  - Content can be reused by reference
  - Content can exist independently of its use within another element

DDI XML Schemas and Main Structures

DDI 3 Main Structures and Concepts

- XML Schemas
  - a file ending with .xsd which describes the structure of an XML file using that schema
- DDI Modules
  - Major functional set of XML Schemas in DDI
- DDI Schemes
  - A maintainable structure found in some DDI modules
- DDI Profiles
  - A means of describing what you use or support in DDI

XML Schemas

- archive
- comparative
- conceptualcomponent
- datacollection
- dataset
- dcelements
- DDIprofile
- ddi-xhtml11
- ddi-xhtml11-model-1
- ddi-xhtml11-modules-1
- group
- inline_ncube_recordlayout
- instance
- logicalproduct
- ncube_recordlayout
- physicaldataproduct
- physicalinstance
- proprietary_record_layout
- reusable
- simpledc20021212
- studyunit
- tabular_ncube_recordlayout
- xml
- set of xml schemas to support xhtml

Types of DDI XML Schemas

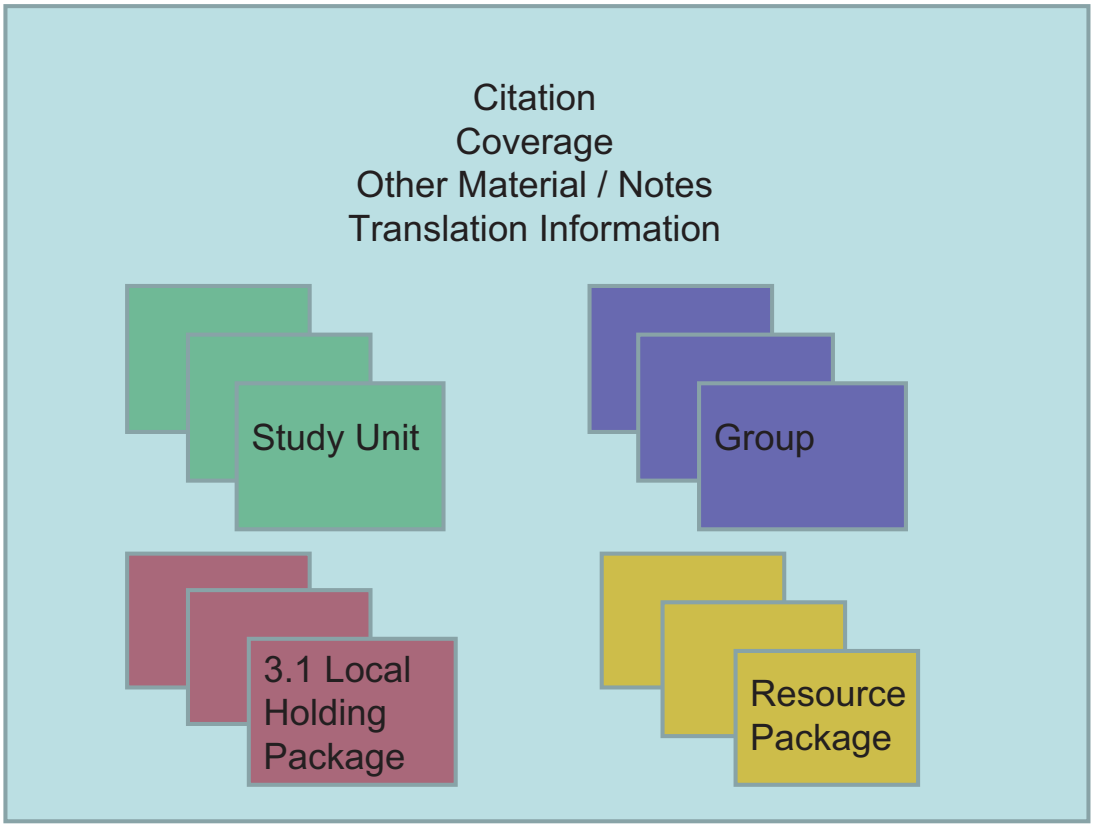
- Packaging / Structural Modules
- Scheme-Based Modules (contain maintainable schemes)
- Non-Scheme-Based Modules
- Sub-Modules (used exclusively in other modules)
- External XML Schemas (eg, Dublin Core)
- Reusable (commonly needed components)

Types of DDI XML Schemas

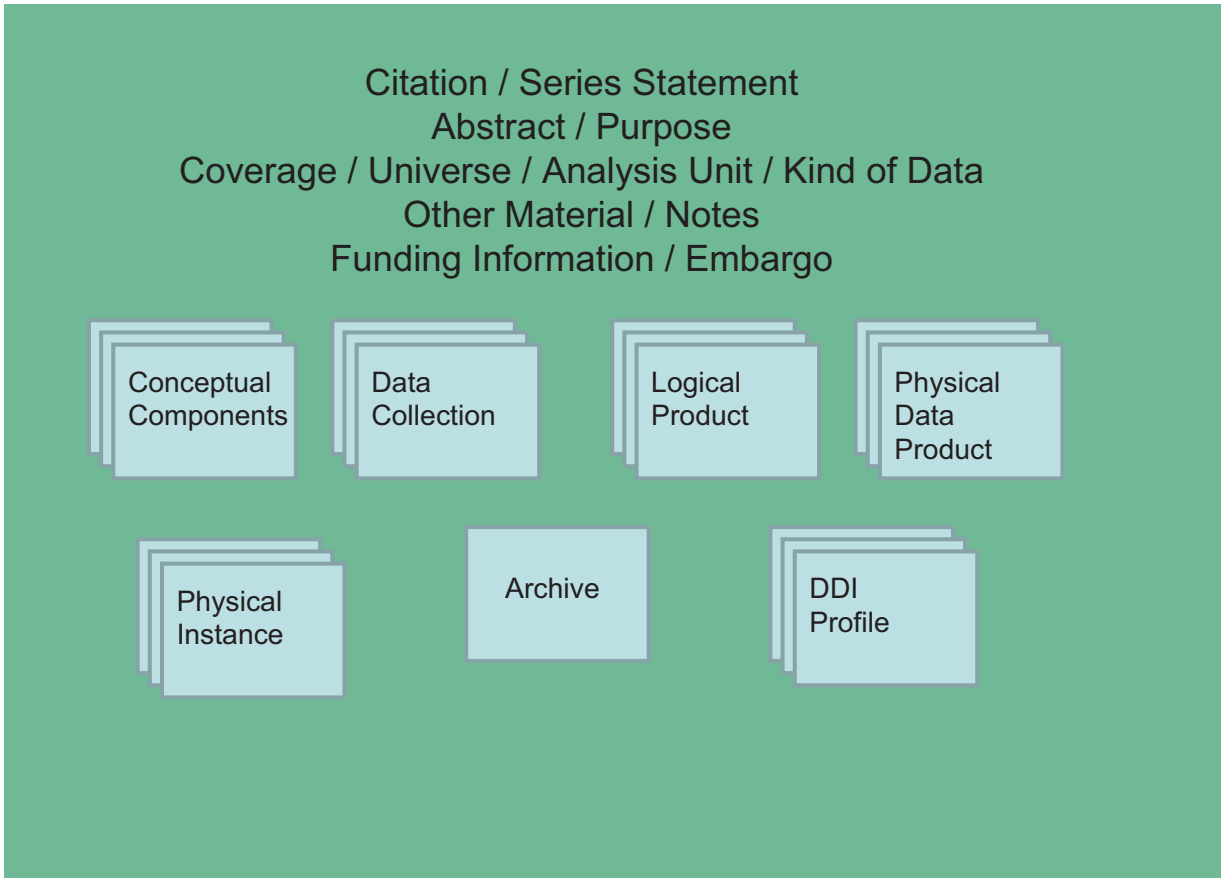
- Packaging / Structural
- Scheme-Based (contain maintainable schemes)
- Non-Scheme-Based
- Sub-Modules (used exclusively in other modules)
- External XML Schemas
- Reusable (commonly needed components)

- DDI Instance
- Study Unit
- Group with top levels:
  - Group
  - Resource Package
  - Local Holding Package

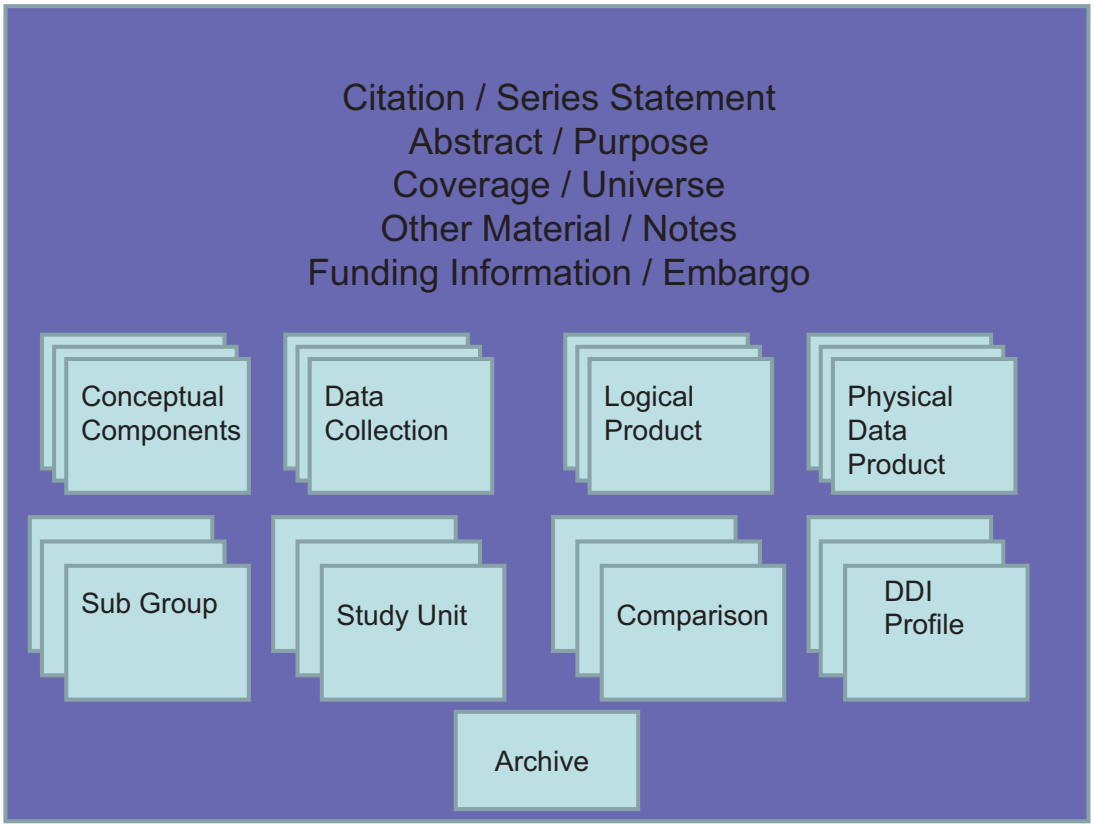
DDI Instance



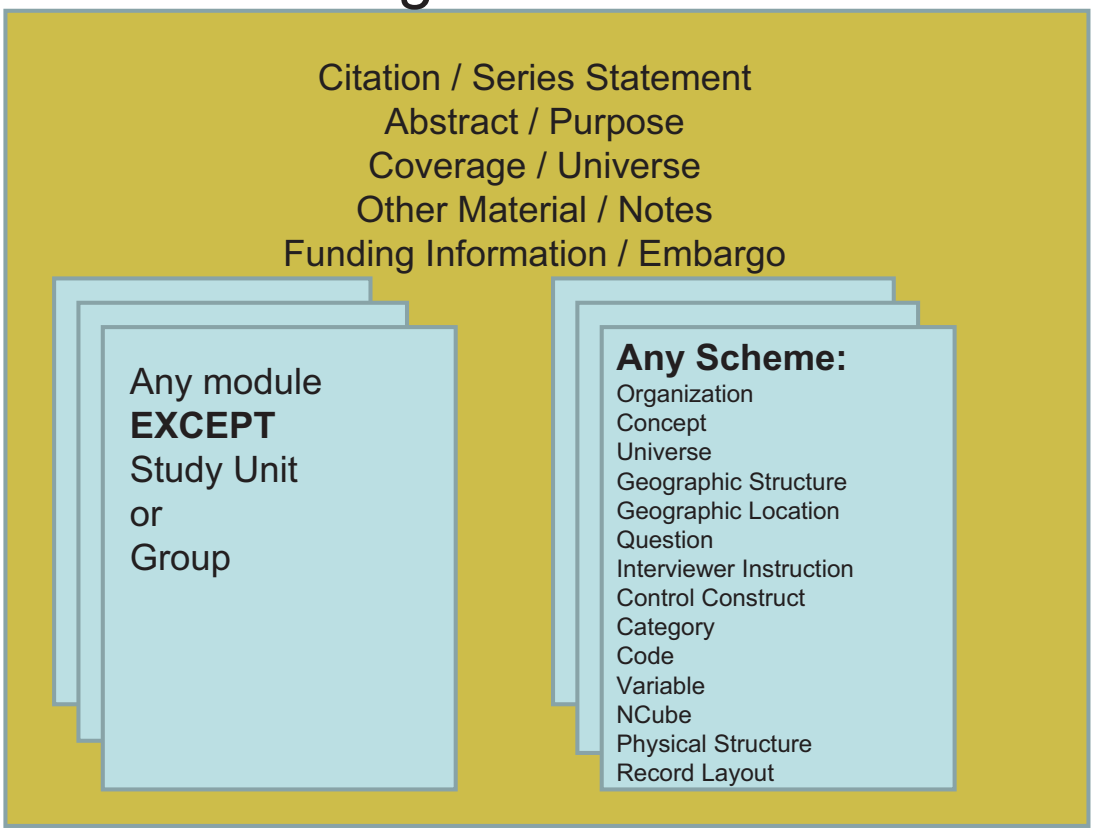
Study Unit



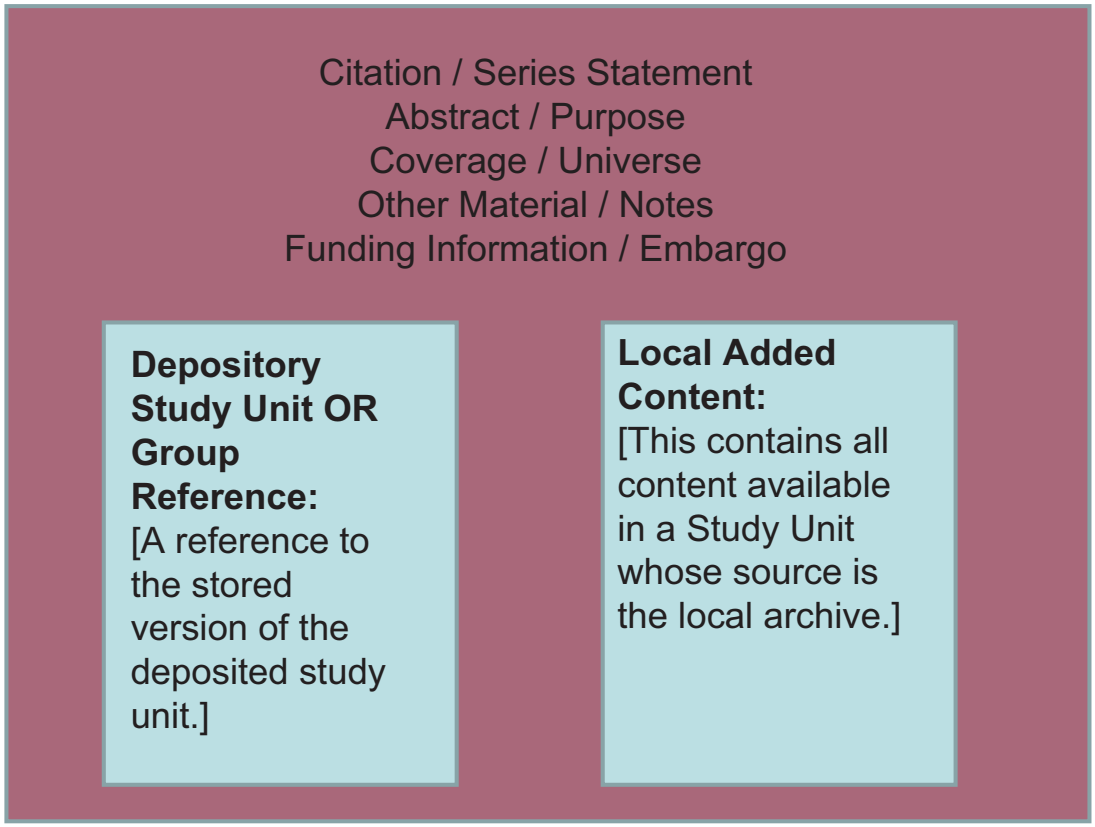
Group



Resource Package



3.1 Local Holding Package



Study Unit

- Study Unit
  - Identification
  - Coverage
    - Topical
    - Temporal
    - Spatial
  - Conceptual Components
    - Universe
    - Concept
    - Representation (optional replication)
  - Purpose, Abstract, Proposal, Funding
- Identification is mapped to Dublin Core and basic Dublin Core is included as an option
- Geographic coverage mapped to FGDC / ISO 19115
  - bounding box
  - spatial object
  - polygon description of levels and identifiers
- Universe Scheme, Concept Scheme
  - link of concept, universe, representation through Variable
  - also allows storage as a ISO/IEC 11179 compliant registry

Group

- Group
  - Up-front design of groups – allows inheritance
  - Ad hoc (“after-the-fact”) groups – explicit comparison using comparison maps for Universe, Concept, Question, Variable, Category, and Code
- Resource Package
  - Allows packaging of any maintainable item as a resource item
- Local Holding Package
  - Allows attachment of local information to a deposited study without changing the version of the study unit itself

Types of DDI XML Schemas

- Packaging / Structural
- Scheme-Based (contain maintainable schemes)
- Non-Scheme-Based
- Sub-Modules (used exclusively in other modules)
- External XML Schemas
- Reusable (commonly needed components)

- archive
  - OrganizationScheme
- datacollection
  - QuestionScheme
  - ControlConstructScheme
  - InterviewerInstructionScheme
- conceptualcomponent
  - ConceptScheme
  - UniverseScheme
  - GeographicStructureScheme
  - GeographicLocationScheme
- logicalproduct
  - CategoryScheme
  - CodeScheme
  - VariableScheme
  - NCubeScheme
- physicaldataproduct
  - PhysicalStructureScheme
  - RecordLayoutScheme

Archive

- An archive is whatever organization or individual has current control over the metadata
- Contains persistent lifecycle events
- Contains archive specific information
  - local identification
  - local access constraints
- Contains Organization information either in-line or by reference

Conceptual Component

- Concept Scheme
  - Describes concepts and concept groups
  - Describes data element concepts
- Universe
  - Provides for hierarchical structure of universes found in the data
- Geographic Structure Scheme
  - Describes geographic levels and their relationships
- Geographic Location Scheme
  - Describes geographic locations and their relationship to geographic levels
  - Can link to shape files or maps

Data Collection

- Methodology
- Collection event
- Question Scheme
  - Question
  - Response domain
- Control Construct Scheme
- Interviewer Instructions
- Instrument
- Processing Event
  - weighting, cleaning, control, data appraisal
  - Coding instructions
- Question and Response Domain designed to support question banks
  - Question Scheme is a maintainable object
- Organization and flow of questions, statements and instructions
  - Used to drive systems like CASES and Blaise
- Coding Instructions
  - Reuse by Questions, Variables, and comparison

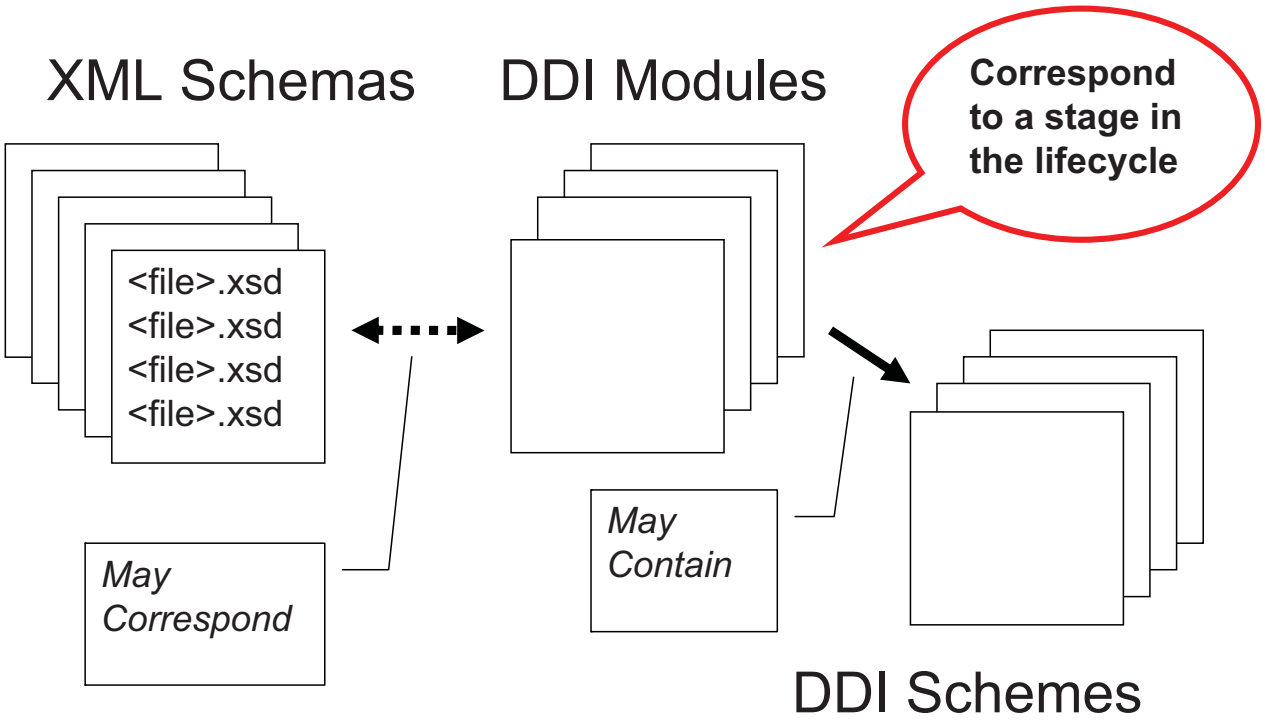
Logical Product

- Category Schemes
- Code Schemes
- Variables
- NCubes
- Variable and NCube Groups
- Data Relationships
- Categories are used as both question response domains and Code Schemes
- Codes are used as both question response domains and variable representations
- Variables link representations to concepts and universes by reference
- NCubes are built from variables (dimensions, measures, and attributes)
  - Map directly to SDMX structures
  - More generalized to accommodate legacy data

Physical Data Product

- Physical Structure Scheme
  - Links to Logical Record in Data Relationships
  - Provides Gross Record Structure information
  - Allows for definition of default values (missing, decimal places, decimal indicator, etc.)
- Record Layout Scheme
  - Identifies type of record layout
  - Links to Variable or NCube Coordinate
  - Description of physical storage structure
    - in-line, fixed, delimited or proprietary
    - means of locating a data item within a record

XML Schemas, DDI Modules, and DDI Schemes



DDI 3 Lifecycle Model and Related Modules

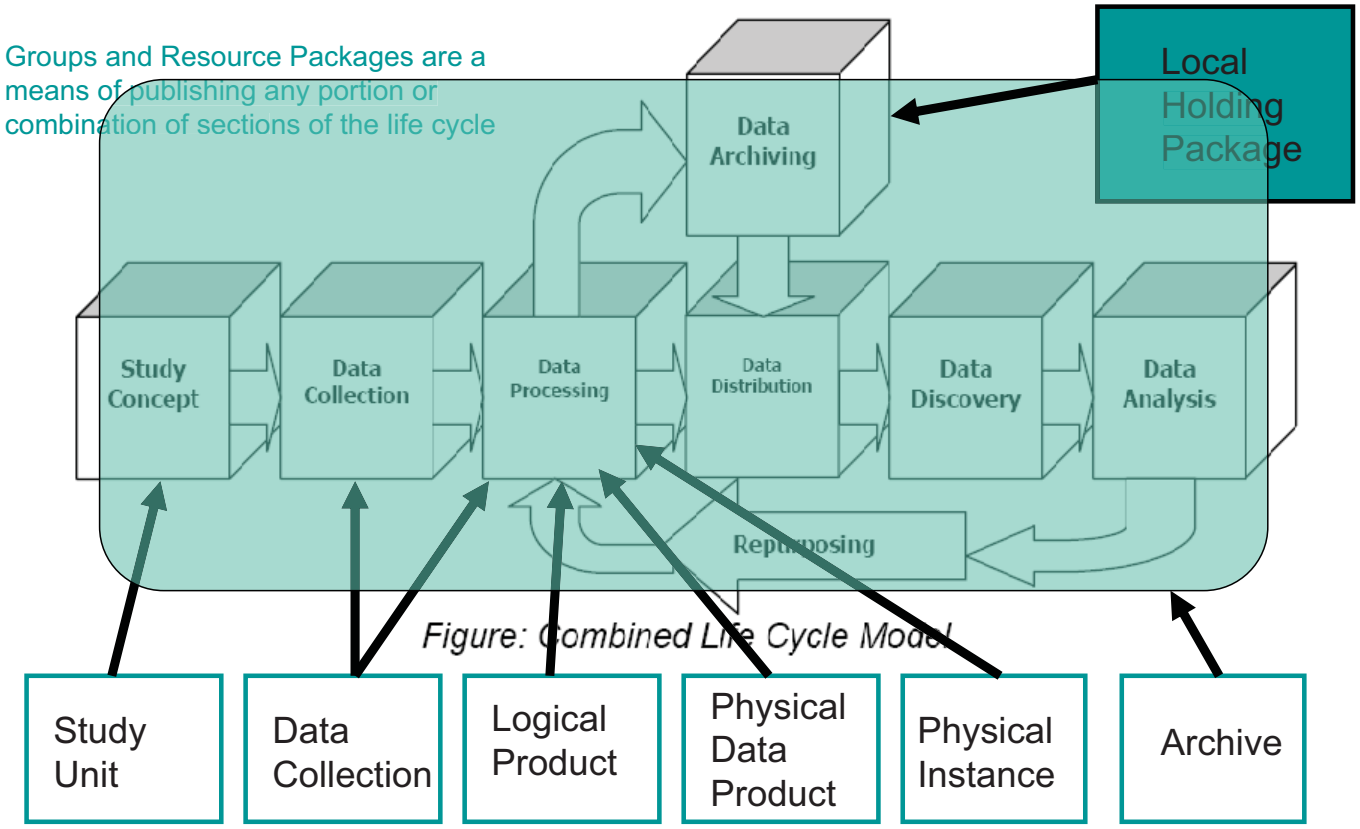


Figure: Combined Life Cycle Model

Types of DDI XML Schemas

- Packaging / Structural
- Scheme-Based (contain maintainable schemes)
- Non-Scheme-Based
- Sub-Modules (used exclusively in other modules)
- External XML Schemas
- Reusable (commonly needed components)

- physicalinstance
- comparative
- DDIprofile

Physical Instance

- One-to-one relationship with a data file
- Citation for the DATA file itself
- Fingerprint(s) for the data file
- Gross file characteristics (check sums, etc.)
- Coverage constraints
- Variable and category statistics

Comparative

- Pair-wise mapping of two objects
- Currently limited to mapping the following:
  - Concept
  - Universe
  - Question
  - Variable
  - Category
  - CodeScheme

DDI's "Meta-Module"

- One module is unlike all of the others in DDI – the DDI Profile
- This is a "meta-module" – it talks about how the DDI 3 is being used by a specific application or organization
  - We do not go into great detail
  - Be aware that Profiles exist however: developers love them!

DDI Profiles

- The DDI Profile module lets you describe which fields you use in your institution's flavor of DDI
  - It is useful for performing machine validation of received instances
  - It is useful documentation for human users
- You provide a set of information for each element allowed in a complete DDI instance
  - If it is used or not used
  - If optional fields (per the XML schema) are required
- Provides the ability to describe DDI Templates
  - Element AlternateName, Description and Instructions
  - Required, default, fixed values

```
<DDIProfile xmlns="ddi:profile:3_1"
   id="DDIProfileSTUDYNO">
   <XPathVersion>1.0</XPathVersion>
   <DDINamespace>3.1</DDINamespace>
   <XMLPrefixMap>
      <XMLPrefix>s</XMLPrefix>
      <XMLNamespace>ddi:studyunit:3_1</XMLNamespace>
   </XMLPrefixMap>
   <Used path="/DDIInstance/VersionResponsibility"/>
   <Used path="/DDIInstance/Citation/Title“/>
   <Used path= required="true" "/DDIInstance/Citation/Creator">
      <AlternateName>Author</AlternateName>
   <Used path="/DDIInstance/StudyUnit/Citation/Title"/>
.....
   <NotUsed path="/DDIInstance/StudyUnit/FundingInformation"/>
</DDIProfile>
```

Types of DDI XML Schemas

- Packaging / Structural
- Scheme-Based (contain maintainable schemes)
- Non-Scheme-Based
- Sub-Modules (used exclusively in other modules)
- External XML Schemas
- Reusable (commonly needed components)

Used in physicaldataproduct:

- inline_ncube_recordlayout
- ncube_recordlayout
- tabular_ncube_recordlayout
- dataset
- proprietary

Types of DDI XML Schemas

- Packaging / Structural
- Scheme-Based (contain maintainable schemes)
- Non-Scheme-Based
- Sub-Modules (used exclusively in other modules)
- External XML Schemas
- Reusable (commonly needed components)

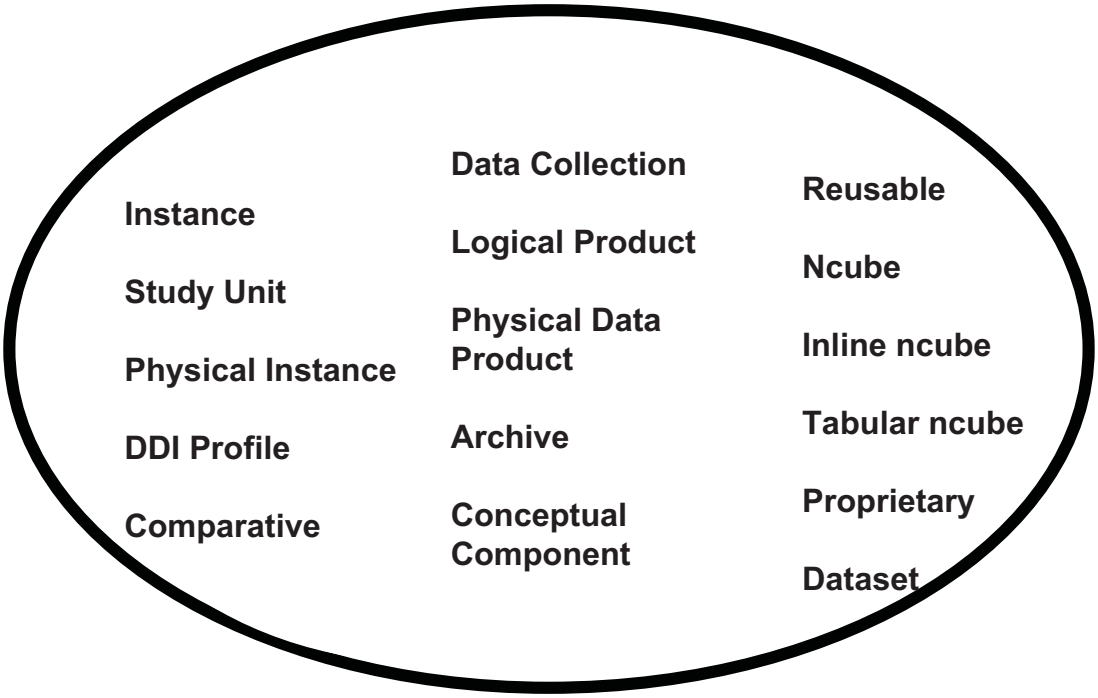
- dcelements
- simpledc20021212
- ddi-xhtml11
- ddi-xhtml11-model-1
- ddi-xhtml11-modules-1
- folder full of xml schemas to support xhtml
- xml

Types of DDI XML Schemas

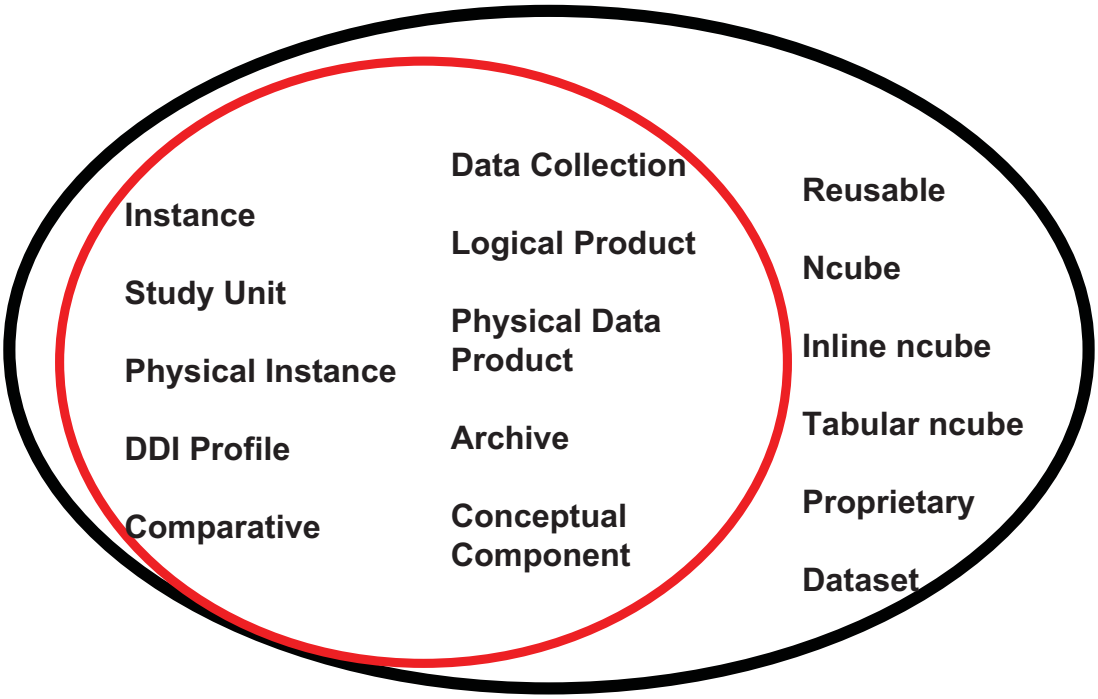
- Packaging / Structural
- Scheme-Based (contain maintainable schemes)
- Non-Scheme-Based
- Sub-Modules (used exclusively in other modules)
- External XML Schemas
- Reusable (commonly needed components)

- Reusable
  - defines elements and complex elements used by the various XML schemas
    - Label
    - Description
    - Notes
    - etc.

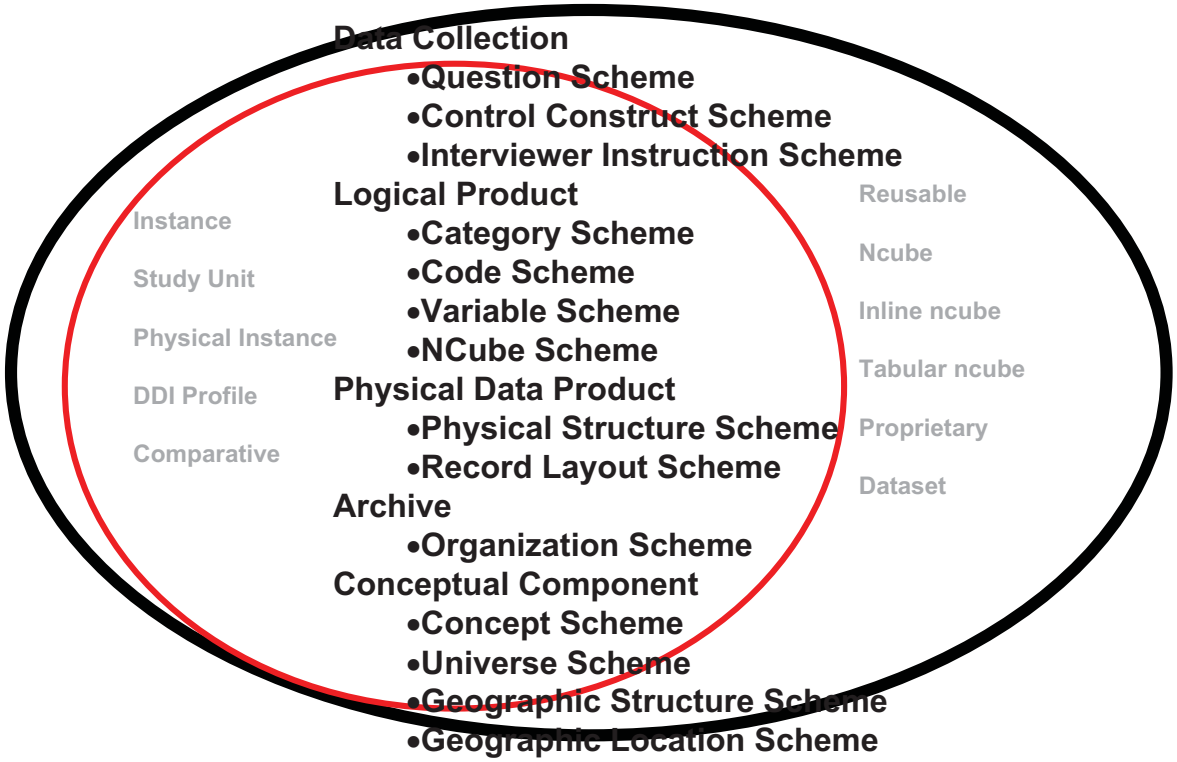
XML Schemas, DDI Modules, and DDI Schemes



XML Schemas, DDI Modules, and DDI Schemes



XML Schemas, DDI Modules, and DDI Schemes



Reminder: DDI Modules and Schemes

- DDI has two important structures:
  - "Modules"
  - "Schemes"
- A module is a package of metadata corresponding to a stage of the lifecycle or a specific structural function
- A scheme is a list of reusable metadata items of a specific type
- Many DDI modules contain DDI schemes

Why Schemes?

- You could ask “Why do we have all these annoying schemes in DDI?”
- There is a simple answer: reuse!
- DDI 3 supports the concept of metadata registries (eg, question banks, variable banks)
- DDI 3 also needs to show specifically where something is reused
  - Including metadata by reference helps avoid error and confusion
  - Comparison is explicit

Designed to Support Registries

- A “Registry” is a catalog of metadata resources
- Resource package
  - Structure to publish non-study-specific materials for reuse
- Extracting specified types of information in to schemes
  - Universe, Concept, Category, Code, Question, Instrument, Variable, etc.
- Allowing for either internal or external references
  - Can include other schemes by reference and select only desired items
- Providing Comparison Mapping
  - Target can be external harmonized structure

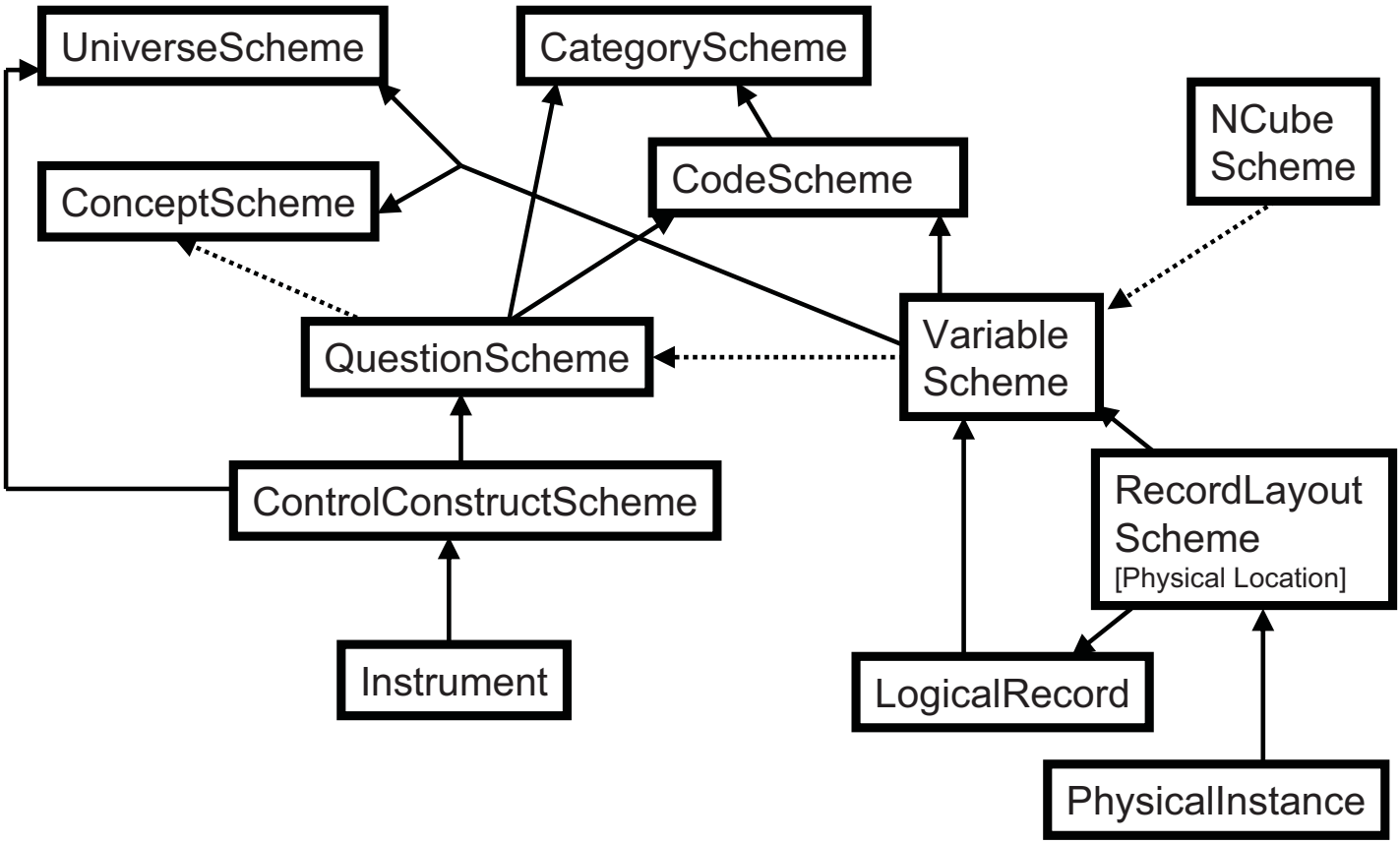
DDI Schemes

- Brief overview of what DDI schemes are and what they are designed to do including:
  - Purpose of DDI Schemes
  - How a DDI Study is built using information held in schemes

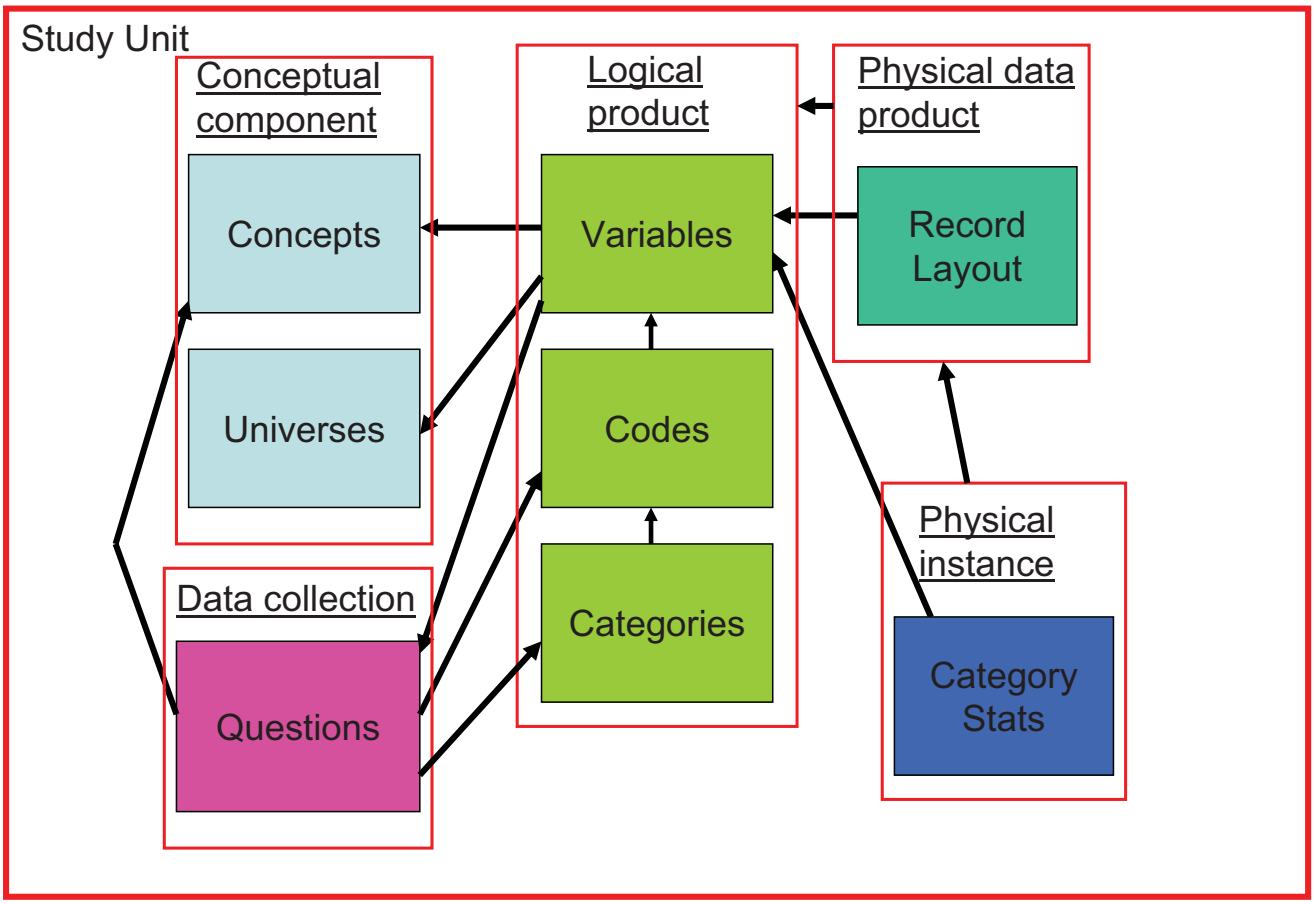
DDI Schemes: Purpose

- A maintainable structure that contains a list of versionable things
- Supports registries of information such as concept, question and variable banks that are reused by multiple studies or are used by search systems to location information across a collection of studies
- Supports a structured means of versioning the list
- May be published within Resource Packages or within DDI modules
- Serve as component parts in capturing reusable metadata within the life-cycle of the data

Building from Component Parts



American National Election Survey Example: Schematic



Technical Features

Core Features of DDI 3

- This section looks at some of the core features of DDI 3 taken as a whole:
  - Identifiables, Versionables, Maintainables
  - Referencing
  - Other materials
- These features support clear and persistent references to DDI elements inside and outside of a DDI instance as well as non-DDI based materials

Rationale

- Because several organizations are involved in the creation of a set of metadata throughout the lifecycle flow:
  - Rules for maintenance, versioning, and identification must be universal
  - Reference to other organization's metadata is necessary for re-use – and *very* common

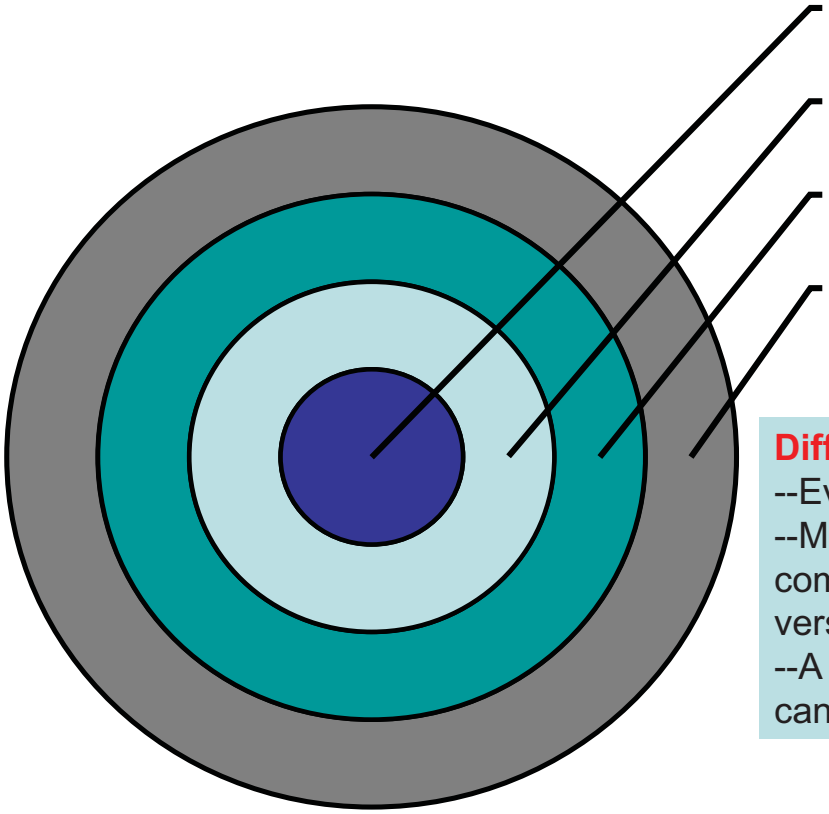
Maintenance Rules

- A maintenance agency is identified by a reserved code based on its domain name (similar to it's website and e-mail)
  - There is a register of DDI agency identifiers which we will look at later in the course
- Maintenance agencies own the objects they maintain
  - Only they are allowed to change or version the objects
- Other organizations may reference external items in their own schemes, but may not change those items
  - You can make a copy which you change and maintain, but once you do that, you own it!

Maintainable, Versionable, and Identifiable

- DDI 3.0 places and emphasis on re-use
  - This creates *lots* of inclusion by reference!
  - This raises the issue of managing change over time
- The Maintainable, Versionable, and Identifiable scheme in DDI was created to help deal with these issues
- An *identifiable object* is something which can be referenced, because it has an ID
- A *versionable object* is something which can be referenced, and which can change over time – it is assigned a version number
- A *maintainable object* is something which is maintained by a specified agency, and which is versionable and can be referenced – it is given a maintenance agency

Basic Element Types

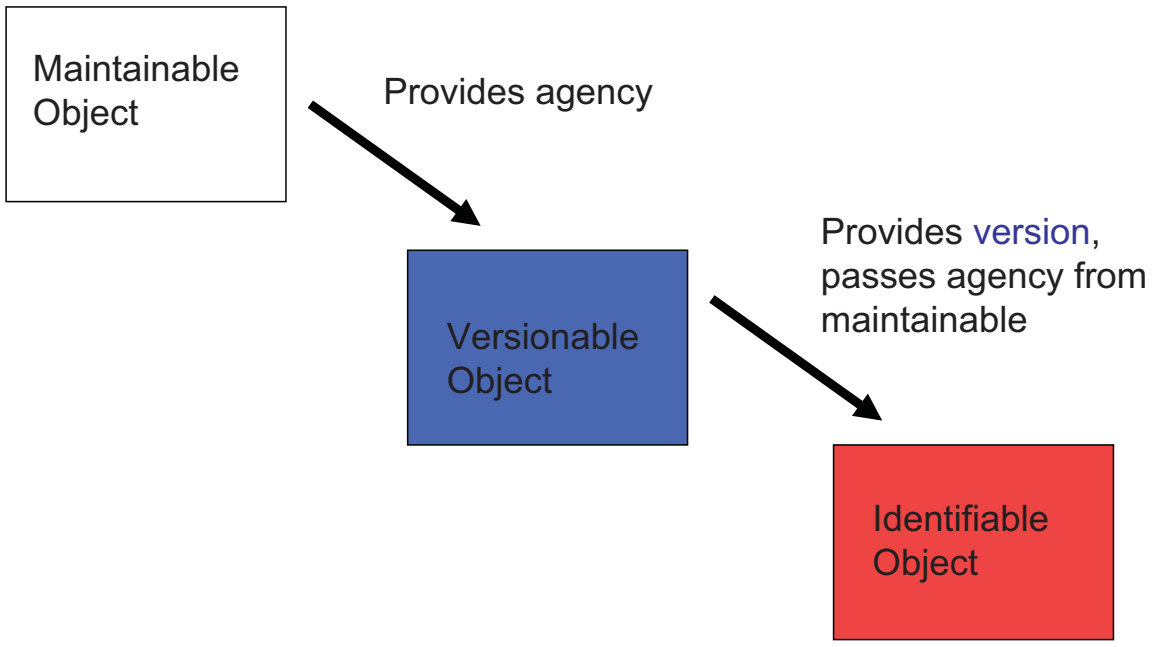


Differences from 2.1
--Every element is NOT identifiable
--Many individual elements or complex elements may be versioned
--A number of complex elements can be separately maintained

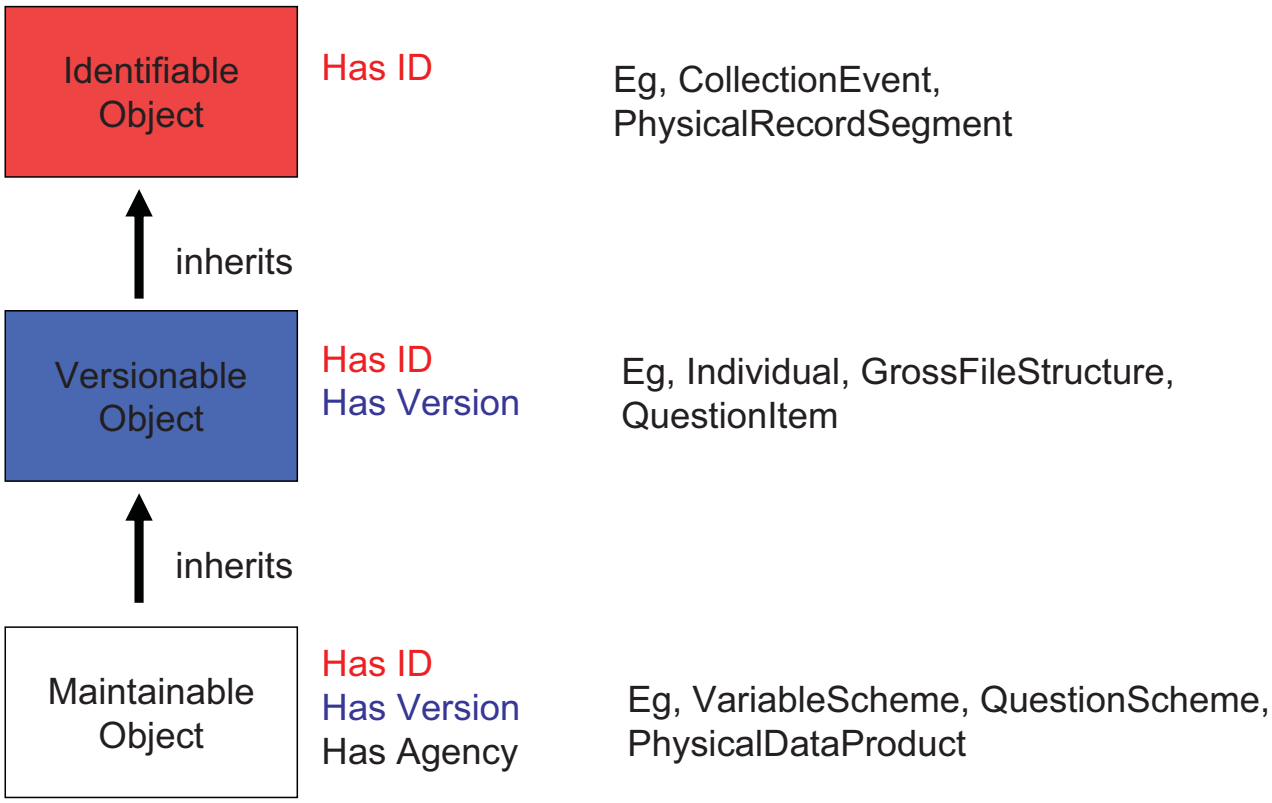
Inheritance of Agency and Version

- In DDI 3 XML instances, identifiables and versionables live in maintainable schemes or modules
  - All of the children of the scheme inherit that scheme's agency
  - If identifiables live inside of a versionable, the identifiables inherit the version number of the versionable
- All of these objects always *implicitly* have an agency, a version, and an ID
- This becomes clear in the way DDI 3 identifiers are structured

In the DDI Instance



In the Object Model…



Versioning and Maintenance

- There are three classes of objects:
  - Identifiable (has ID)
  - Versionable (has version and ID)
  - Maintainable (has agency, version, and ID)
- Very often, identifiable items such as Codes and Variables are maintained in parent schemes

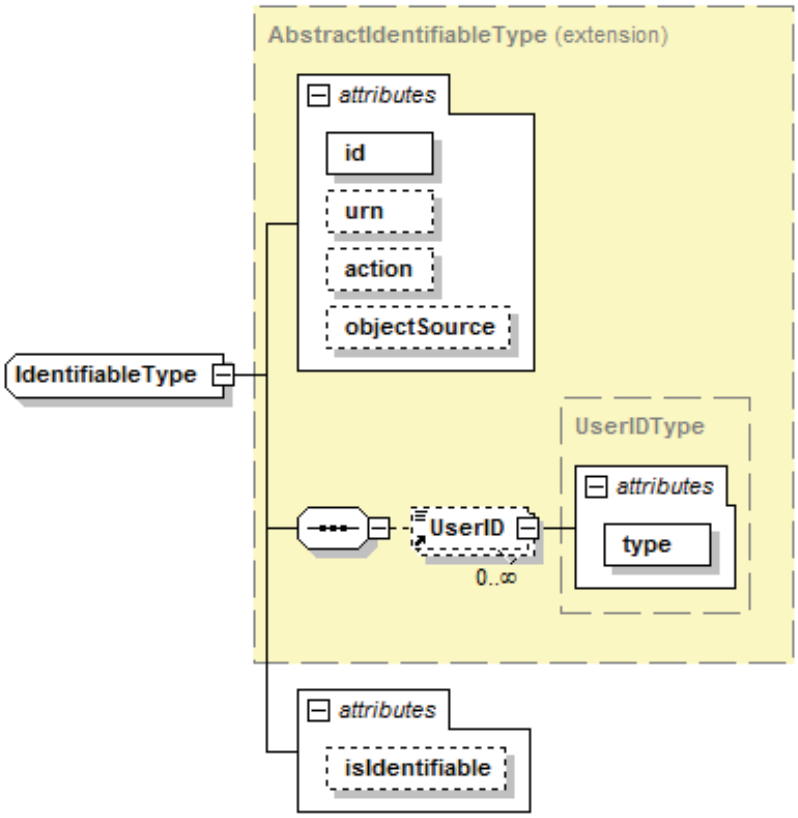
What Does This Mean?

- As different pieces of metadata move through the lifecycle, they will change.
  - At a high level, “maintainable” objects represent packages of re-usable metadata passing from one organization to another
  - Versionable objects represent things which change as they are reviewed within an organization or along the lifecycle
  - Identifiable things represent metadata which is reused at a granular level, typically within maintainable packages
- The high-level documentation lists out all maintainables, versionables, and identifiables in a table

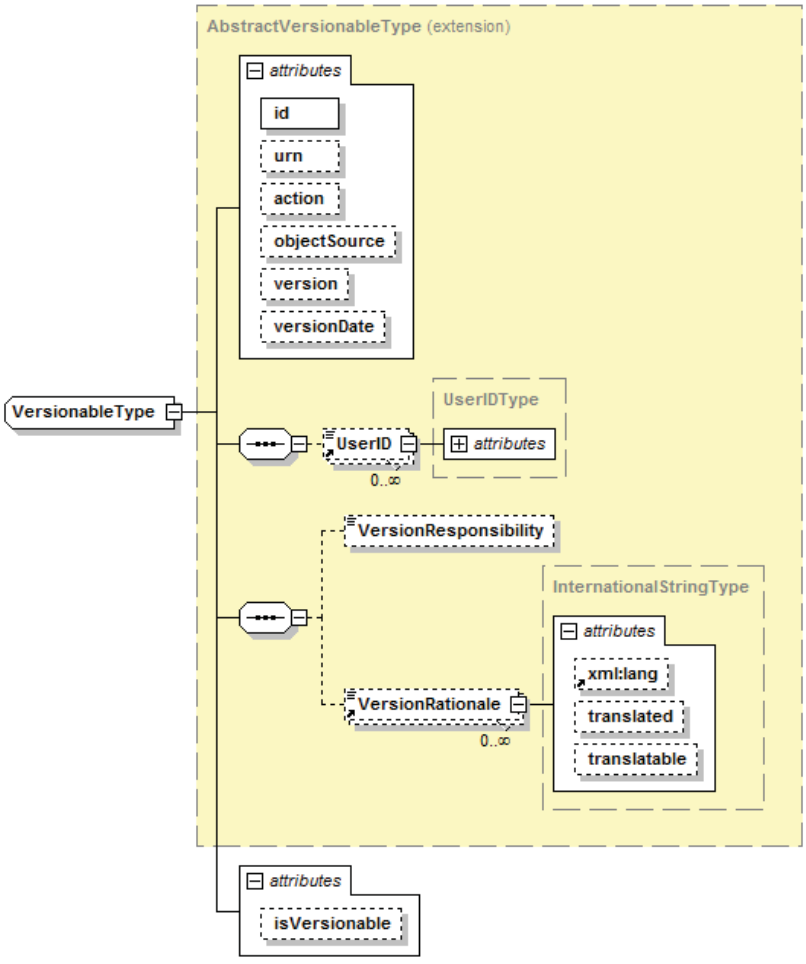
DDI 3.1 Identifiers

- There are two ways to provide identification for a DDI 3 object:
  - Using a set of XML fields
  - Using a specially-structured URN
- The structured URN approach is preferred
  - URNs are a very common way of assigning a universal, public identifier to information on the Internet
  - However, they require explicit statement of agency, version, and ID information in DDI 3
- Providing element fields in DDI 3 allows for much information to be defaulted
  - Agency can be inherited from parent element
  - Version can be inherited or defaulted to “1.0.0”

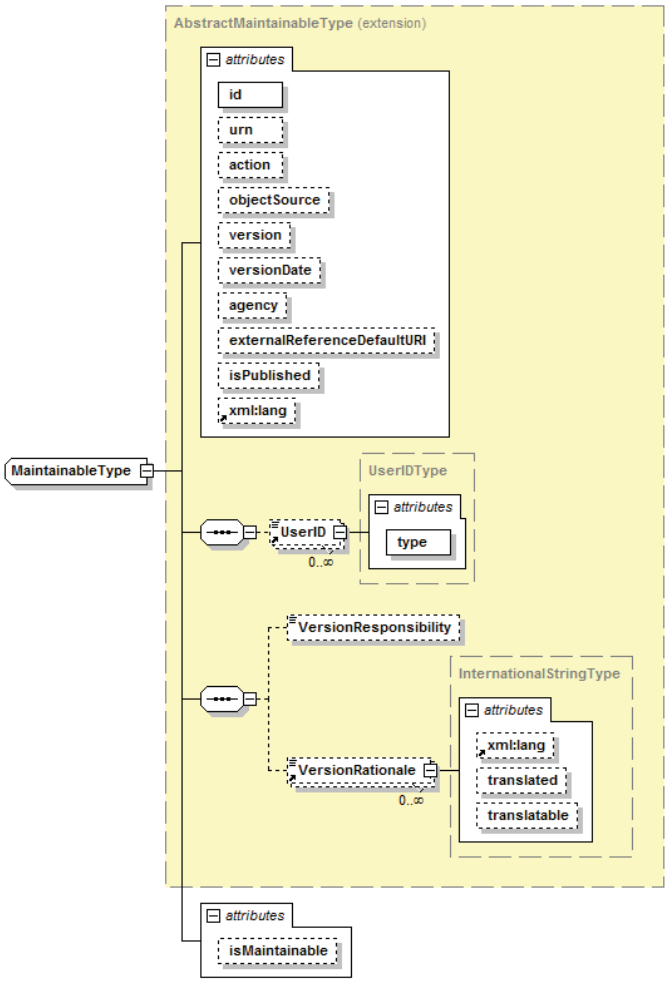
Identification Types - Identifiable



Identification Types: Versionable



Identification Types: Maintainable



Parts of the Identification Series

- Identifiable Element
  - Identifier:
    - ID
    - Identifying Agency
    - Version
    - Version Date
    - Version Responsibility
    - Version Rationale
    - UserID
    - Object Source
- Variable
  - Identifier:
    - V1
    - us.mpc
    - 1.1.0 [default is 1.0.0]
    - 2007-02-10
    - Wendy Thomas
    - Spelling correction

DDI Identifiers: Elements

- **Typical appearance (identifiable):**

```
<pd:DataItem id="AB347"
  isIdentifiable="true">
    …
</pd:DataItem>
```

- **Typical appearance (versionable):**

```
<l:Variable id="V1" version="1.1.0"
  versionDate="2007-02-12"
  isVersionable="true">
  <r:VersionResponsibility>Wendy
  Thomas</r:VersionResponsibility>
   <r:VersionRationale>Spelling
  Correction</r:VersionRationale>
    …
</l:Variable >
```

DDI Identifiers: Elements (cont.)

- **Typical appearance (maintainable):**

```
<l:VariableScheme id="VarSch01" agency
  ="us.mpc" version="1.4.0"
  versionDate="2009-02-12"
  isMaintainable="true">
    …
</l:VariableScheme>
```

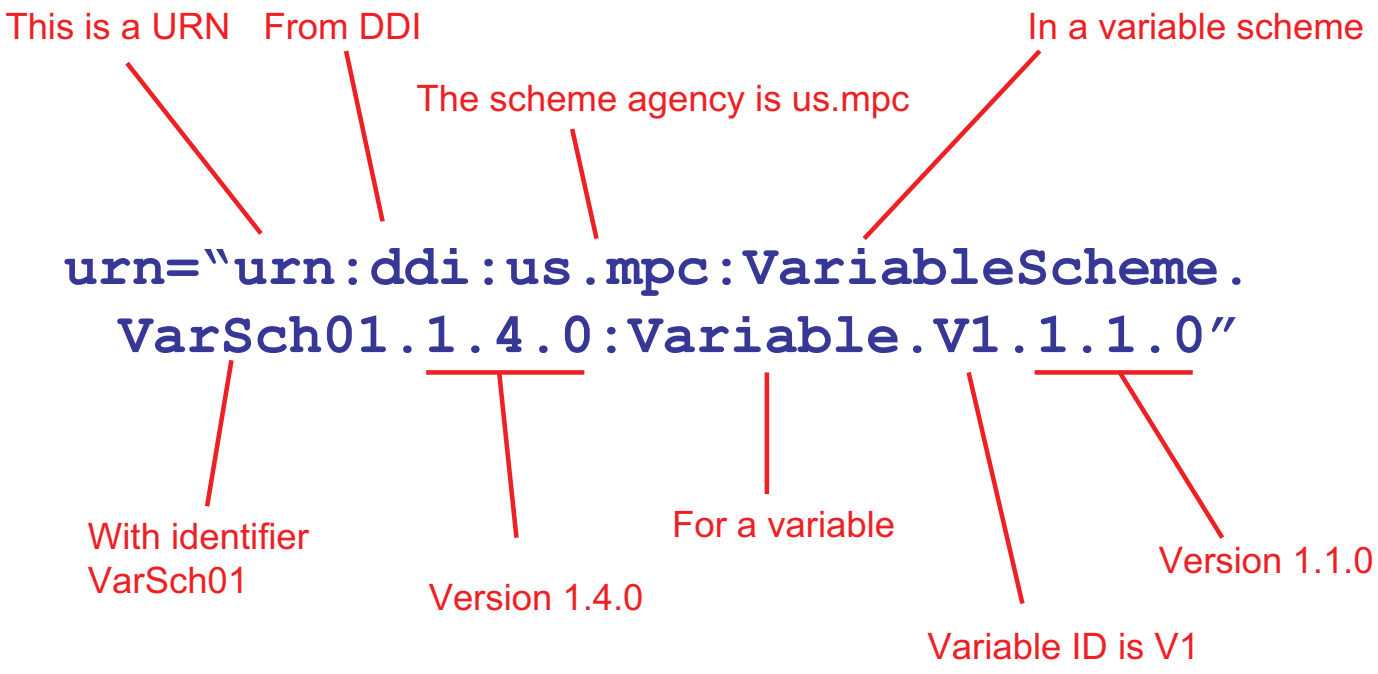
- Note that version and agency may be defaulted/inherited, which means they do not need to be supplied in the local element
  - In a simple example, they are given once for the whole study
  - The object type is determined by the containing element

The URN

urn="urn:ddi:us.mpc:VariableScheme.VarSch01.1.4.0:Variable.V1.1.1.0"

- Declares that its a ddi element
- Gives the identifying agency
- Tells the type of the element that is the parent maintainable including ID and version number
- Tells the type of the element itself including ID and version number
  - Note that the element ID must be unique within its maintainable object rather than within the agency
- There are generic tools for resolving URNs
  - They are mapped to local URLs

URN Detailed Example



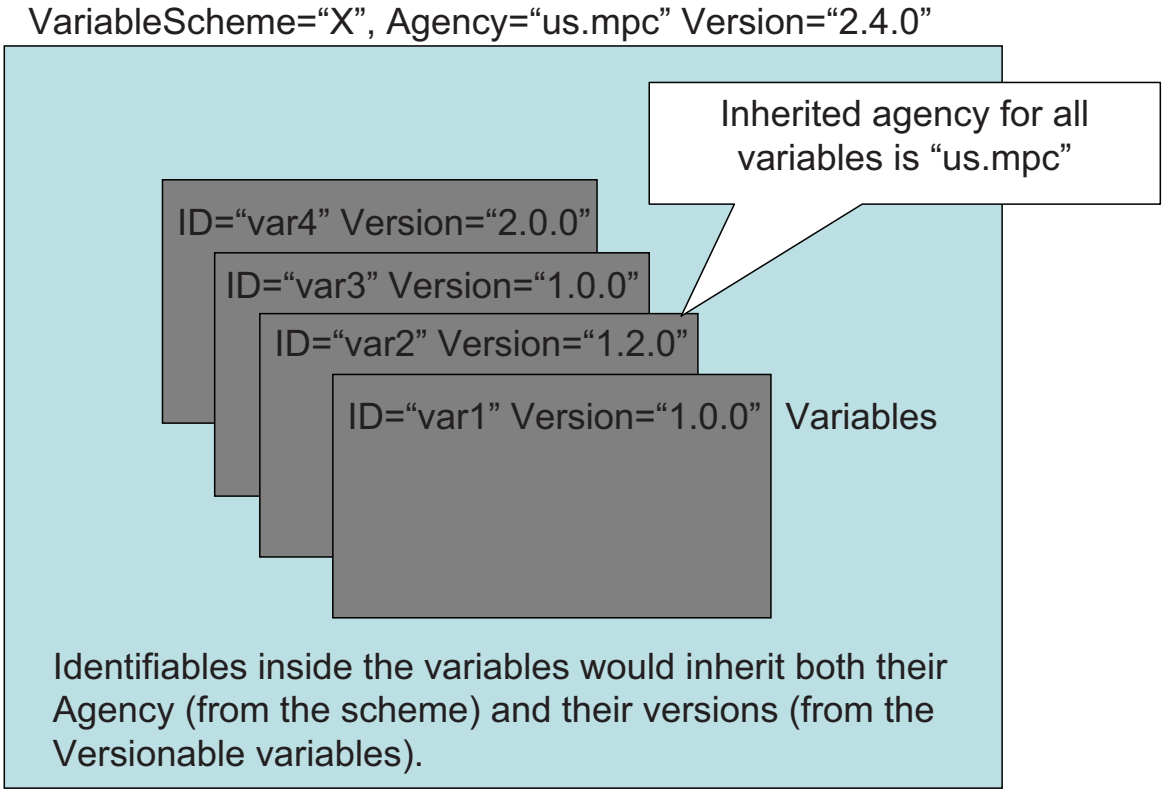
Changes to URN Syntax

- Due to implementation experiences with DDI 3.0, the URN syntax was heavily revised for version 3.1
- What we present here is the 3.1 version of the URN syntax

Identifiable Rules

- Identifiers are assigned to each identifiable object, and are unique within their maintained parent scheme
- Identifiable objects inherit their version from their containing versionable parent (if any)
- Identifiable objects inherit their maintaining agency from the maintainable object they live in

Inheriting Identifying Fields



Versioning Rules

- If an object changes in any way, its version changes
- This will change the version of any containing maintainable object
- Typically, objects grow and are versioned as they move through the lifecycle
- Versions inherit their agency from the maintainable scheme they live in

Versioning Across the DDI 3 Lifecycle Model

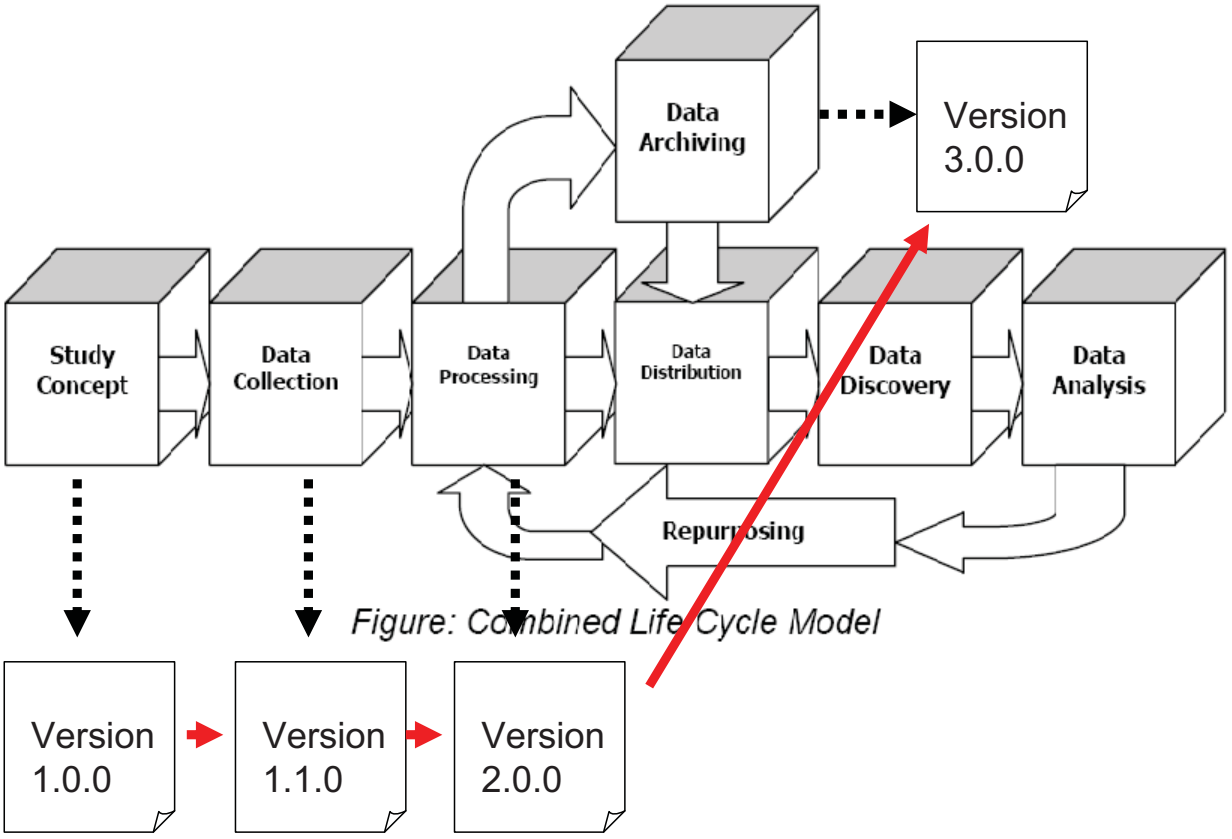


Figure: Combined Life Cycle Model

Versioning: Changes

ConceptScheme X
V 1.0.0
- Concept A v 1.0.0
- Concept B v 1.0.0
- Concept C v 1.0.0

references

ConceptScheme X
V 1.1.0
- Concept A v 1.1.0
- Concept B v 1.0.0
- Concept C v 1.1.0
Add:
Concept D v 1.0.0

references

ConceptScheme X
V 2.0.0
- Concept A v 1.2.0
- Concept B v 1.0.0
- Concept C v 1.2.0
- Concept D v 1.1.0
Add:
Concept E v 1.0.0

references

ConceptScheme X
V 3.0.0
- Concept D v 1.1.0
- Concept E v 1.0.0

references

Note: You can also reference entire schemes and make additions

Publication in DDI

- There is a concept of “publication” in DDI which is important for maintenance, versioning, and re-use
- Metadata is “published” when it is exposed outside the agency which produced it, for potential re-use by other organizations or individuals
  - Once published, agencies must follow the versioning rules
  - Internally, organizations can do whatever they want before publication
- Note that an “agency” can be an organization, a department, a project, or even an individual for DDI purposes
  - It must be described in an Organization Scheme, however!
- There is an attribute on maintainable objects called “isPublished” which must be set to “true” when an object is published (it defaults to “false”)

Non-XML Structures

- DDI may not be always be maintained within an XML expression of the contents
- Most systems will use various forms of relational data bases to hold metadata content
- The information in the data base must support the identification, versioning and reference rules of DDI if it is to support reuse of metadata

Referencing

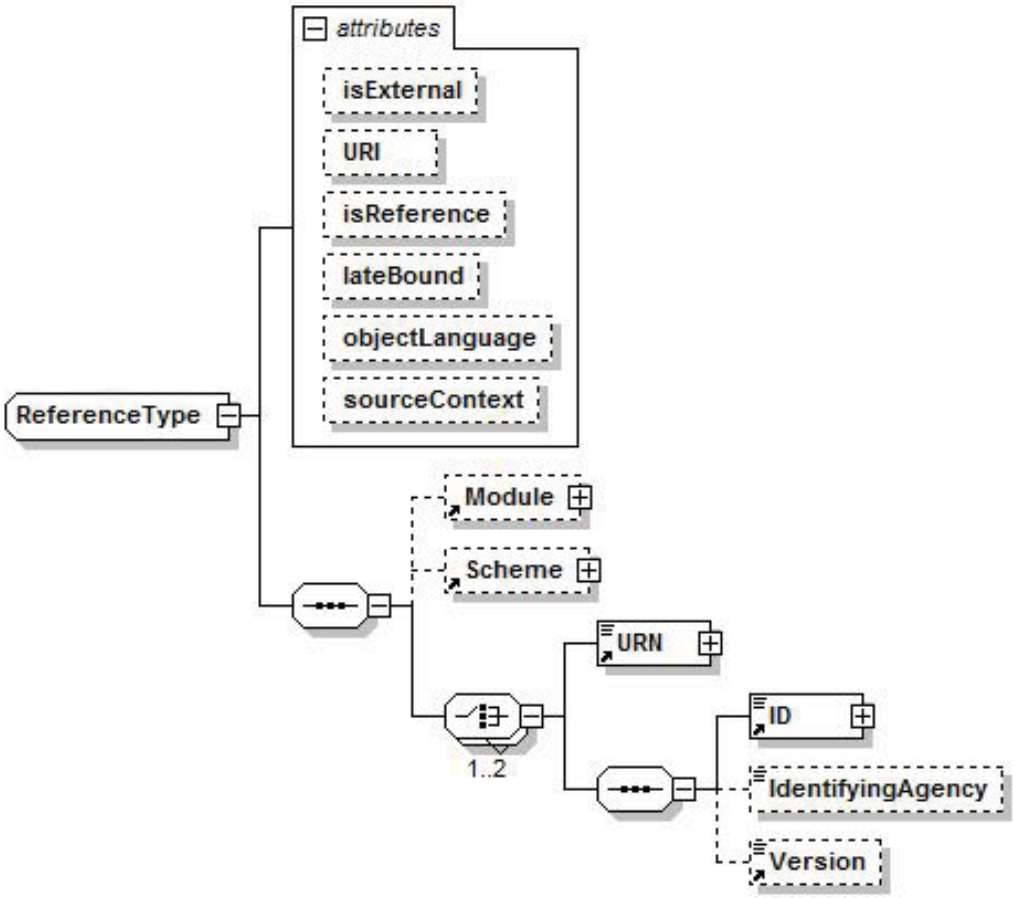
- When referencing an object, you must provide:
  - The maintenance agency
  - The identifier
  - The version
- Often, these are inherited from a maintained scheme
  - This is part of their identification

DDI Internal References

- References in DDI may be within a single instance or across instances
  - Metadata can be re-packaged into many different groups and instances
- Identifiers must provide:
  - The containing module (optional)
    - Agency, ID, and Version
  - The containing maintainable (a scheme)
    - Agency, ID, and Version
  - The identifiable/versionable object within the scheme
    - ID (and version if versionable)
- Like identifiers, DDI references may be using URNs or using element fields

DDI External References

- Change attribute isExternal to “true”
- ALL DDI references to external objects must contain a URN
- This may be accompanied by the same individual elements, but if there is a discrepancy between this information and the URN, the URN will take precedence
- Beginning with DDI 3.1 you can also designate both the objectLanguage and sourceContext if broader than the parent maintainable
  - ObjectLanguage specifies which language to use for display (if more than one is present)
  - sourceContext identifies where the referenced object is coming from, identified with a URN, in cases where a specific object is available from more than one version of a scheme



Reference Examples

- Internal

```
<VariableReference isReference="true"
  isExternal="false" lateBound="false">
 <Scheme isReference="true" isExternal="false"
  lateBound="false">
  <ID>VarSch01</ID>
  <IdenftifyingAgency>us.mpc</IdentifyingAgency>
  <Version>1.4.0</Version>
 </Scheme>
  <ID>V1</ID>
  <IdenftifyingAgency>us.mpc</IdentifyingAgency>
  <Version>1.1.0</Version>
</VariableReference>
```

Reference Examples

- External

```
<VariableReference isReference="true"
  isExternal="true" lateBound="false">
<urn>urn:ddi:us.mpc:VariableScheme
  .VarSch01.1.4.0:Variable.V1.1.1.0
  </urn>
</VariableReference>
```

High level information on identification within major modules

- DDI Instance, Group, Resource Package, Study Unit, and Physical Instance
- Citation
  - Bibliographic identification
- Coverage
  - The who, what, when, and where
- Relational context
  - Other Materials

Citations

- Citations are available in several modules:
  - Instance
  - Study Unit
  - Group
  - Resource Package
  - Physical Instance
- In addition to the standard citation elements you can also include simple Dublin Core elements in their native format

Maintainables without Citations

- All maintainables without citations (i.e., Schemes) and versionable objects that are designed to be held in a registry have a standard means of identification following the ISO/IEC 11179-5 structure:
  - xxxName
  - Label
  - Description

Coverage

- There are three types of Coverage in DDI 3:
  - Spatial (geographic)
  - Temporal (time)
  - Topical (subject and keyword)

Spatial Coverage

- Describe the geographic cover in detail
- References the geographic hierarchies found in the data and which have data summarized at that level
- Provides information on the smallest and largest spatial object type found in the data
- References Geographic Structure and Locations

Bounding Box

<r:BoundingBox>
<r:NorthLatitude>**+76.63**</r:NorthLatitude>
<r:EastLongitude>**-61.48**</r:EastLongitude>
<r:SouthLatitude>**+13.71**</r:SouthLatitude>
<r:WestLongitude>**-177.1**</r:WestLongitude>
</r:BoundingBox>

Description

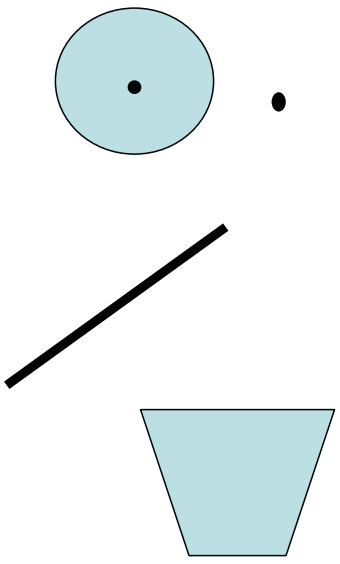
<r:Description translated="false" translatable="true">

<xhtml:p>United States, Region, Division, State, County, County Subdivision, Place, Tract/Block Numbering Area within Place/Remainder within County Subdivision.</xhtml:p>

</r:Description>

Spatial Object

- Point
  - Address
  - Coordinate point
- Line
  - Street
  - Boundary
- Polygon
- Linear Ring
  - Point and radius



Topical Coverage

- Subject
  - A structured set of terms expressing topical coverage of the study content. Examples include: U.S. Library of Congress Subject Headings, Medical Subject Headings (MESH), etc.
- Keyword
  - Unstructured words selected due to frequency of occurrence within a study or terms that may be linked to subjects in external systems such as synonyms or representations of a subject. Example of a keyword “Blue”: this can be a color or a feeling. Searches of keywords do not differentiate.
- Maps to Dublin Core elements

Controlled Vocabularies

- One feature which is now being addressed is the use of controlled vocabularies
  - These occur in many places in the schemas
  - There is a working group looking at these issues and making recommendations
- There is an OASIS standard called “Genericode” which addresses the use of controlled vocabularies in XML standards
  - It allows them to be customized by different user communities
  - It separates them from the versioning of the XML standard itself
- DDI will use Genericode to handle controlled vocabularies in 3.0 and moving ahead

Genericode

- Every place in the DDI XML which uses a controlled vocabulary has three attributes:
  - Codelist ID
  - Codelist Name
  - Codelist Agency Name
  - Codelist version ID
  - Codelist URN
  - Codelist Scheme URN
- These point to an external codelist maintained in Genericode XML
- Controlled vocabularies are not validated not by the normal XML parser using the DDI schemas
  - They are validated after-the-fact using a separate mechanism such as Schematron

International Code Value

- Content
- Attributes:
  - ISO language code (xml:lang) - required
  - Boolean “translated” – default “false”
  - Boolean “translatable” – default “false”
  - Genericode code list identification (codelistID)
  - Agency maintaining the code list (codeListAgency)
  - Version of the code list (codeListVersion)

Subject Example

```
<r:Subject xml:lang="en"
   codeListID="DDI_MeSH"
   codeListAgency="MESH"
   codelistVersion="1.0">
   Arthritis.Rheumatoid
</Subject>
```

Genericode Example

```
<Row>
   <Value ColumnRef="Code">
      <SimpleValue>Arthritis.Rheumatoid</SimpleValue>
   </Value>
   <Value ColumnRef="ParentCode">
      <SimpleValue>Arthritis</SimpleValue>
   </Value>
   <Value ColumnRef="LevelSpecificCode">
      <SimpleValue>Rheumatoid</SimpleValue>
   </Value>
   <Value ColumnRef="Definition">
      <SimpleValue>Inflammatory arthritis, a chronic symptom
      disease, primarily of the joints</SimpleValue>
   </Value>
   <Value ColumnRef="Caption">
      <SimpleValue/>
   </Value>
</Row>
```

Temporal Coverage

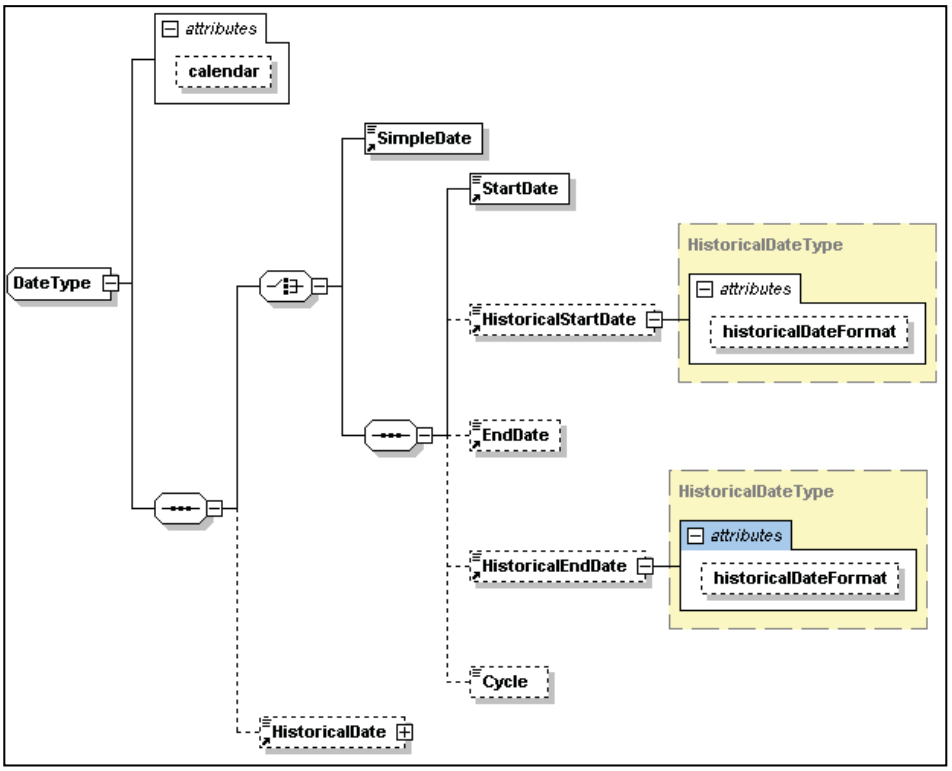
- Set of Reference Dates describing the time period covered by the study and data

```
<Date>
  <StartDate>2009-03-30</StartDate>
  <HistoricalStartDate>
     March 30, 2009
  </HistoricalStartDate>
  <EndDate>2009-04-03</EndDate>
  <HistoricalEndDate>
     April 3, 2009
  </HistoricalEndDate>
</Date>
```

Dates

- ISO required structure
- Combine various year+month+day+time
- SimpleDate
- Date Ranges
- Noting Historical Date structures
- Calendar types
- DateTime Response Domains
  - Declare type used for non-ISO structures

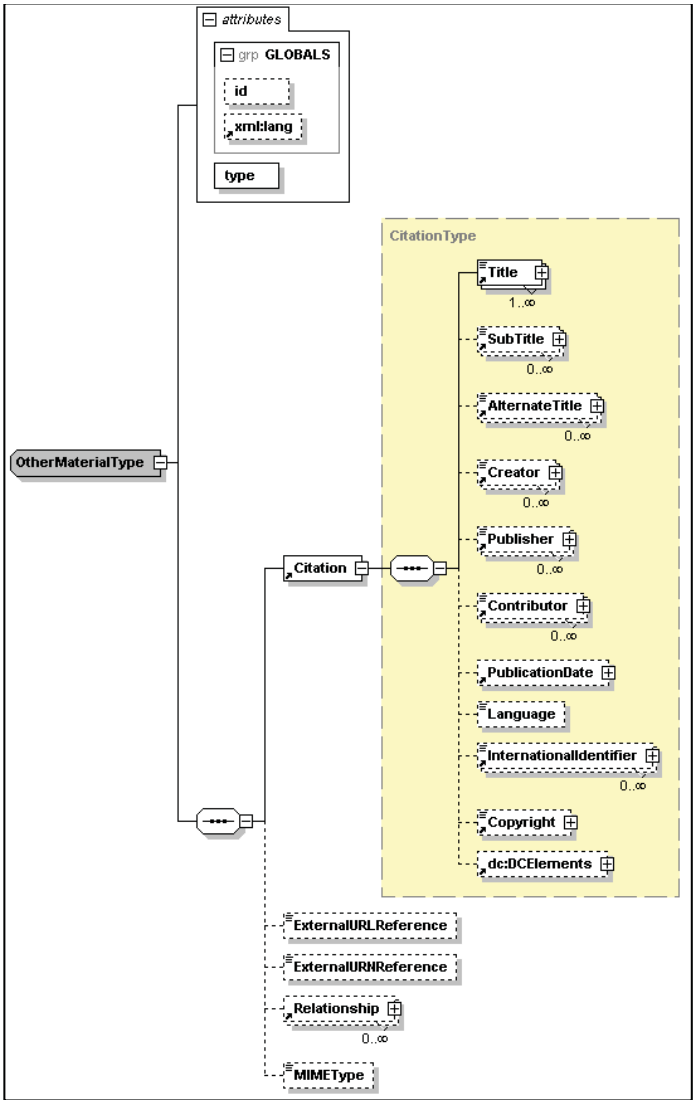
Dates



Other Material

- Available in all major modules
- Can be linked to any element with an ID
- Can by typed with a controlled vocabulary
- The purpose of the Other Material structure is reference and so contains only the basic bibliographic citation and location information if available
- Provides relationship information containing reason for relationship and reference to one or more elements

Other Material 3.0



Other Material DDI 3.1

- Added Segment information
  - Allows designating start and stop information for
    - Text
    - Audio
    - Video
    - XML
  - Allows linking to specific segments of other forms of electronic materials

DDI 3 Anticipated Use

Use Cases

- To provide context for the course, we will introduce some typical/possible uses of DDI 3
- We cannot know which ones will become the most common
  - These cases are partly based on discussions and planned projects
  - Many of these use cases are in active implementation today, or already in production

Use Cases

- Study design/survey instrumentation
- Questionnaire generation/data collection and processing
- Data recoding, aggregation and other processing
- Data dissemination/discovery
- Archival ingestion/metadata value-add
- Question/concept/variable banks
- DDI for use within a research project
- Capture of metadata regarding data use
- Metadata mining for comparison, etc.
- Generating instruction packages/presentations
- Registry data

Study Design/Survey Instrumentation

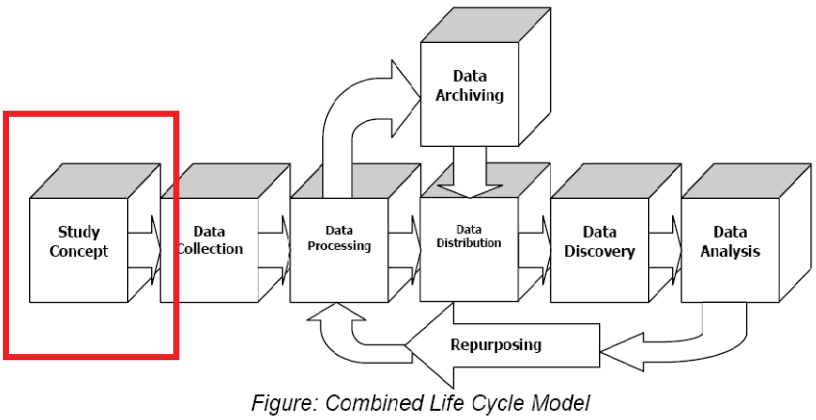
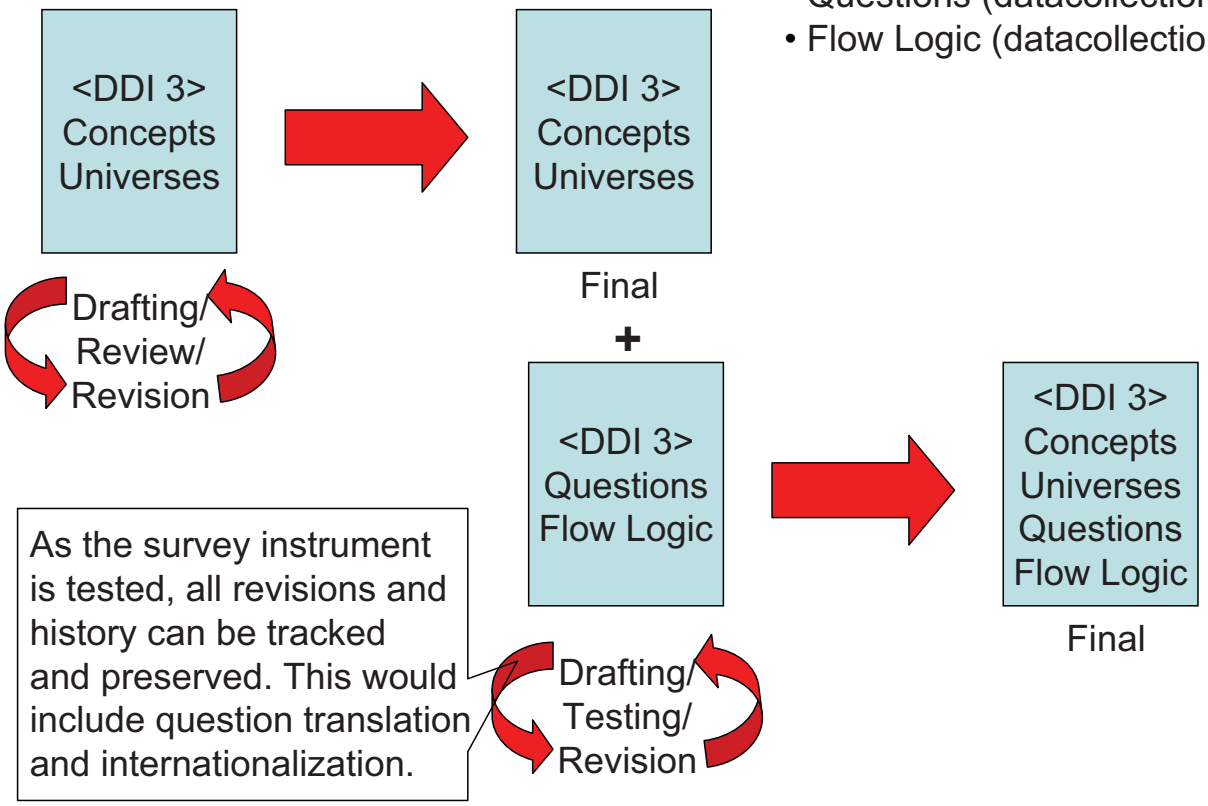


Figure: Combined Life Cycle Model

- This use case concerns how DDI 3 can support the design of studies and survey instrumentation
  - Without benefit of a question or concept bank

Types of Metadata:
- Concepts (conceptual module)
- Universe (conceptual module)
- Questions (datacollection module)
- Flow Logic (datacollection module)



studyunit.xsd
conceptualcomponent.xsd
datacollection.xsd
logicalproduct.xsd

<s:StudyUnit>
required and optional study unit elements plus
<cc:ConceptualComponents>
<cc:ConceptScheme>
<cc:UniverseScheme>
<d:DataCollection>
<d:QuestionScheme>
<d:ControlConstructScheme>
<d:Processing>
<l:LogicalProduct>
<l:CategoryScheme>
<l:CodeScheme>

Questionnaire Generation, Data Collection, and Processing

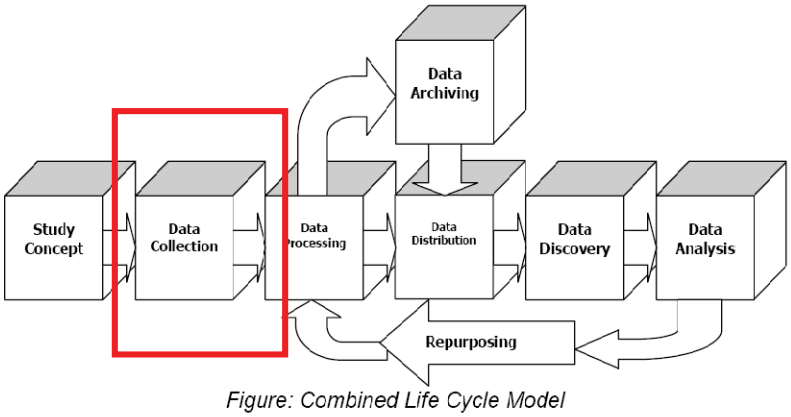
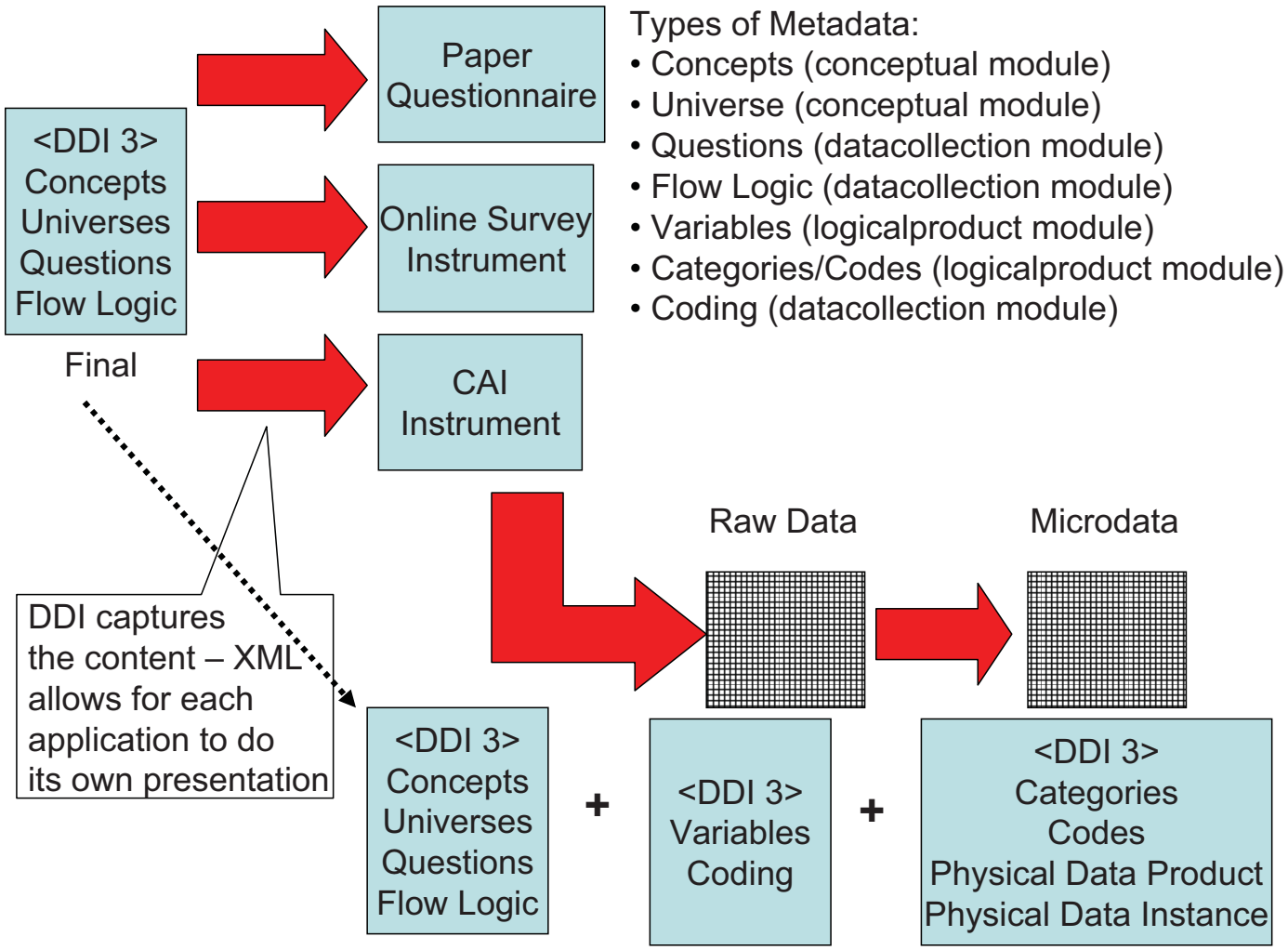


Figure: Combined Life Cycle Model

- This use case concerns how DDI 3 can support the creation of various types of questionnaires/CAI, and the collection and processing of raw data into microdata.



studyunit.xsd
conceptualcomponent.xsd
datacollection.xsd
logicalproduct.xsd
physicaldatastructure.xsd
physicalinstance.xsd

Previous structure PLUS

<l:LogicalProduct>
    <l:DataRelationship>
    <l:VariableScheme>
<pd:PhysicalDataStructure>
    <pd:PhysicalStructureScheme>
    <pd:RecordLayoutScheme>
<pi:PhysicalInstance>

Data Recoding, Aggregation, etc.

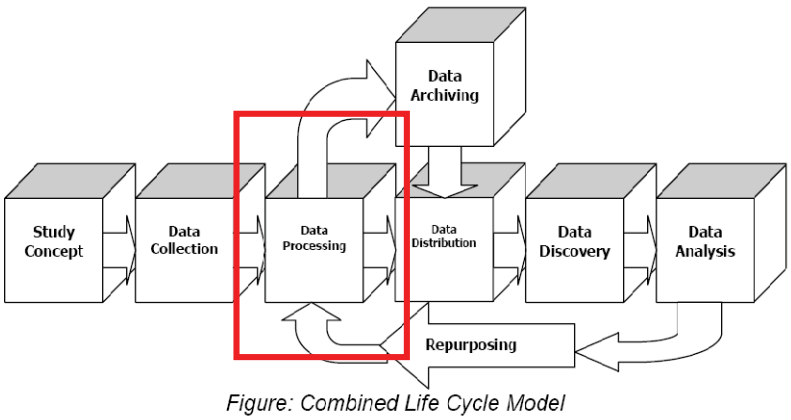
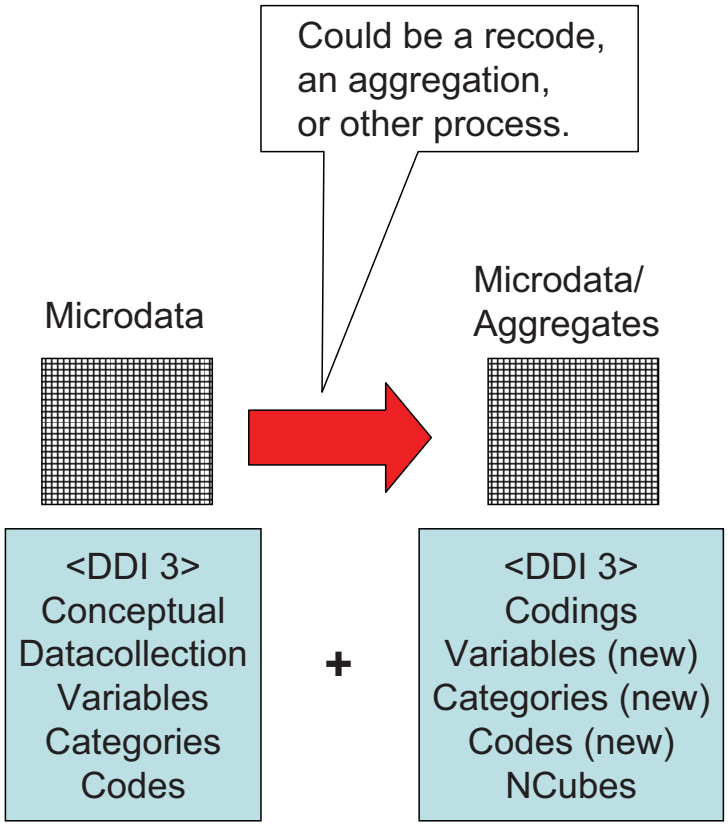


Figure: Combined Life Cycle Model

- This use case concerns how DDI 3 can describe recodes, aggregation, and similar types of data processing.



Initial microdata has:

- Concepts (conceptual module)
- Universes (conceptual module)
- Questions (datacollection module)
- Flow Logic (datacollection module)
- Variables (logicalproduct module)
- Coding (datacollection module)
- Categories (logicalproduct module)
- Codes (logicalproduct module)
- Physical Data Product
- Physical Data Instance

Recode adds:

- More codings (datacollection module)
- New variables
- New categories
- New codes
- NCubes (for aggregation)

studyunit.xsd
conceptualcomponent.xsd
datacollection.xsd
logicalproduct.xsd
physicaldatastructure.xsd
physicalinstance.xsd

ADD to the following schemas

<d:DataCollection> new <d:Processing> elements

<l:LogicalProduct>

<l:NCubeScheme>

<pd:PhysicalDataStructure> new NCube Record Layout

<pi:PhysicalInstance> additional physical instances

Data Dissemination/Data Discovery

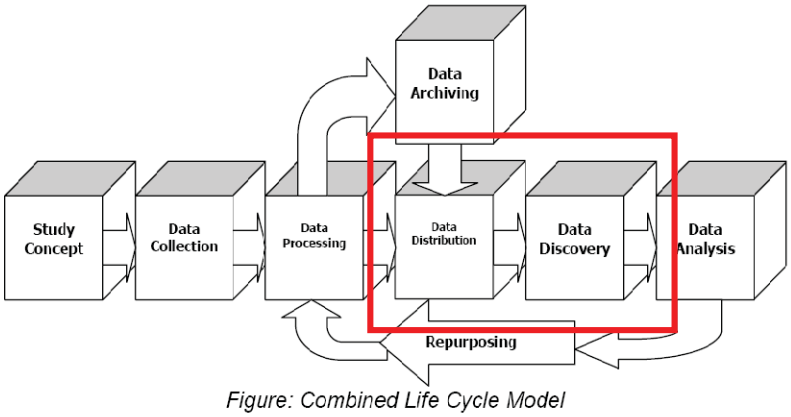
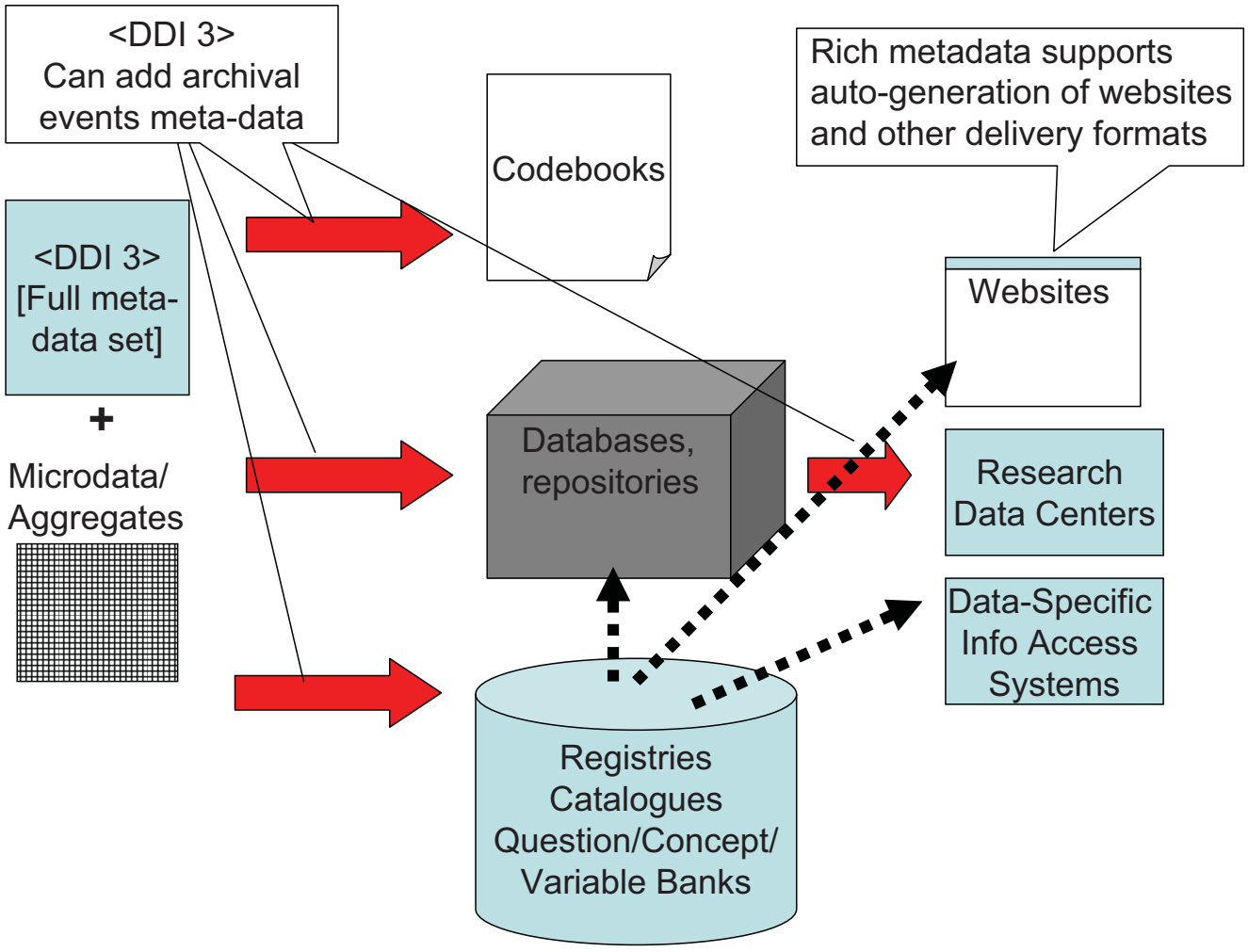


Figure: Combined Life Cycle Model

- This use case concerns how DDI 3 can support the discovery and dissemination of data.



<cc:ConceptScheme>
<cc:UniverseScheme>
<cc:GeographicStructureScheme>
<cc:GeographicLocationScheme>
<d:QuestionScheme>
<d:ControlConstructScheme>
<l:VariableScheme>
<l:CategoryScheme>
<l:CodeScheme>
<pd:PhysicalStructureScheme>
<pd:RecordLayoutScheme>
<a:OrganizationScheme>
<s:StudyUnit> [descriptive content]

- Store as separate resources
- Use content to feed a different registry structure

Archival Ingestion and Metadata Value-Add

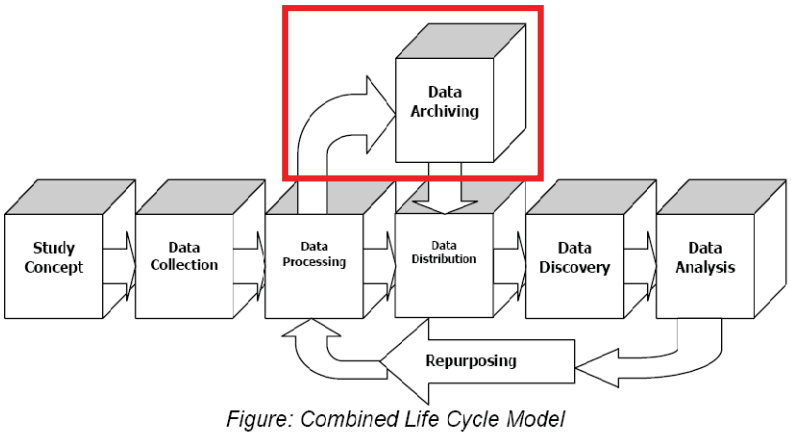
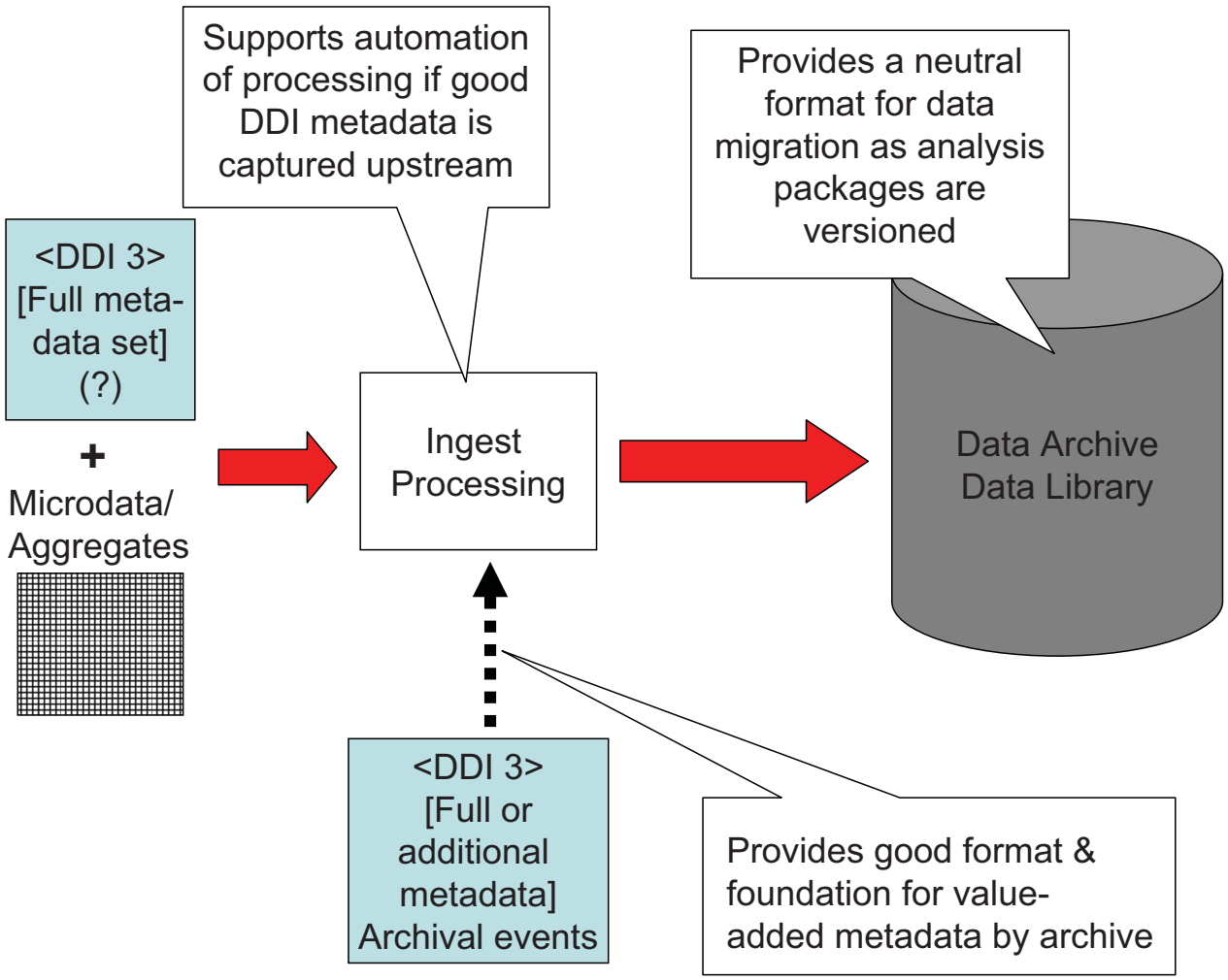


Figure: Combined Life Cycle Model

- This use case concerns how DDI 3 can support the ingest and migration functions of data archives and data libraries.



<g:LocalHoldingPackage>

<s:StudyUnit>

with full content

OR

<g:Group>

with full content

+

<s:StudyUnit>

new value added content

<a:Archive>

<a:LifeCycleEvents>

capture ingest processing events

Question/Concept/Variable Banks

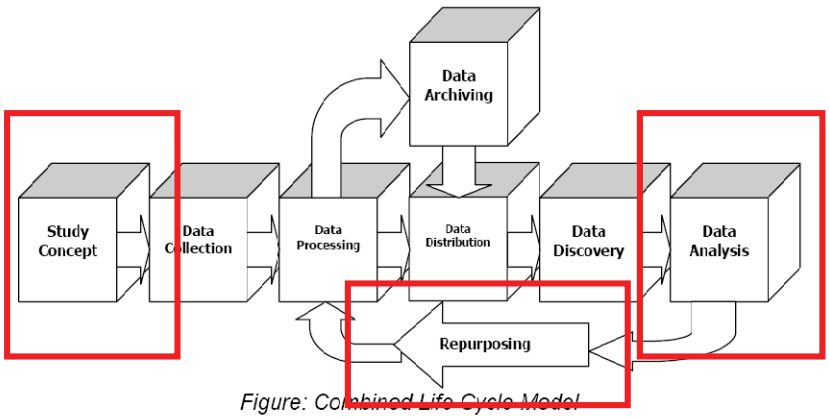
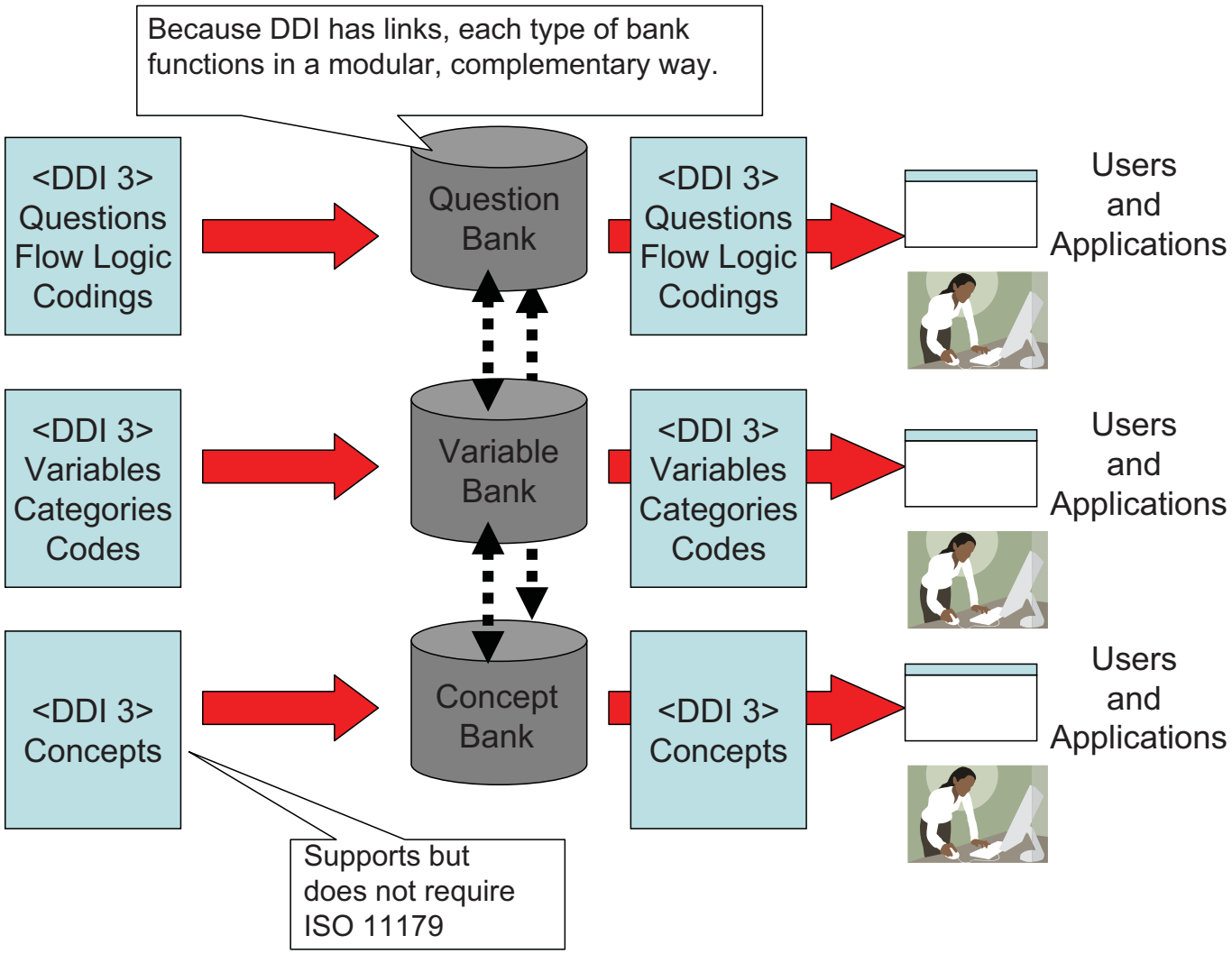


Figure: Combined Life Cycle Model

- This use case describes how DDI 3 can support question, concept, and variable banks. These are often termed “registries” or “metadata repositories” because they contain only metadata – links to the data are optional, but provide implied comparability. The focus is metadata *reuse.*



<g:ResourcePackage>

- Question Bank
  - <d:QuestionScheme>
  - <d:ConstrolConstructScheme>
- Variable Bank
  - <l:CategoryScheme>
  - <l:CodeScheme>
  - <l:VariableScheme>
- Concept Bank
  - <cc:ConceptScheme>

DDI For Use within a Research Project

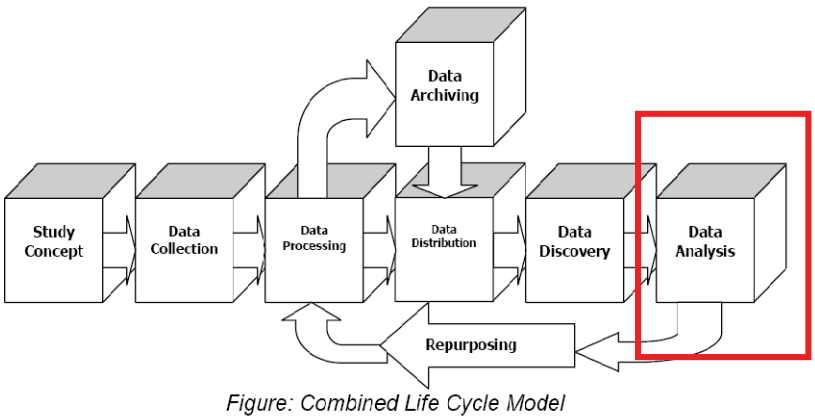
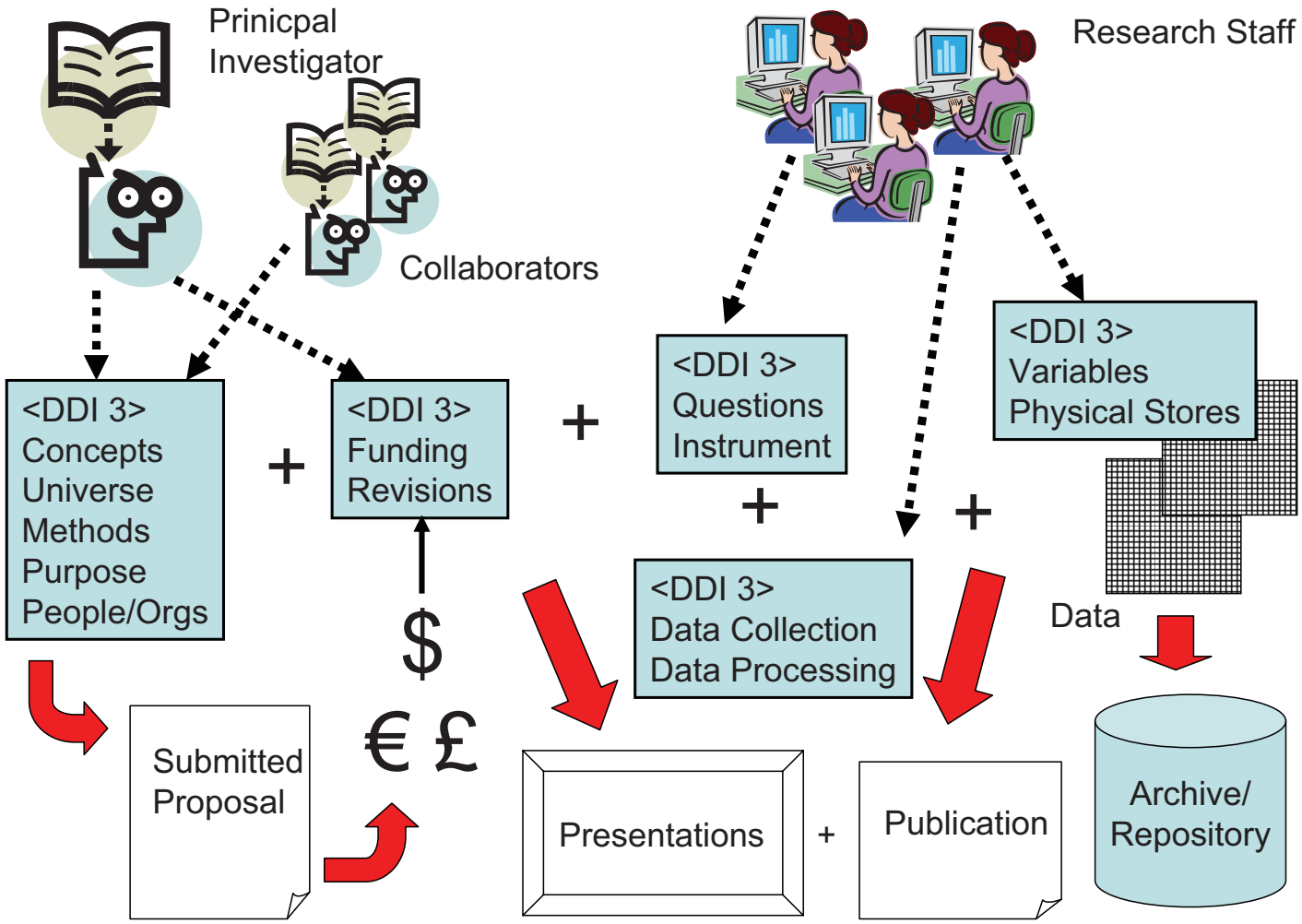


Figure: Combined Life Cycle Model

- This use case concerns how DDI 3 can support various functions within a research project, from the conception of the study through collection and publication of the resulting data.




```
<s:StudyUnit>
    <s:Abstract>
    <s:Purpose>
    <r:FundingInformation>
    <c:ConceptualComponents>
        <cc:Concepts>
        <c:Universe>
    <d:DataCollection>
        <d:Methodology>
        <d:QuestionScheme>
        <d:ControlConstructScheme>
    <l:LogicalProduct>
        <l:DataRelationship>
        <l:CategoryScheme>
        <l:CodeScheme>
        <l:VariableScheme>
    <p:PhysicalDataProduct>
    <pi:PhysicalInstance>
    <a:Archive>
        <a:OrganizationScheme>
```

- Version 1.0.0 Preparing the proposal for funding

<s:StudyUnit>
<s:Abstract>
<s:Purpose>
<r:FundingInformation>
<c:ConceptualComponents>
<cc:Concepts>
<c:Universe>
<d:DataCollection>
<d:Methodology>
<d:QuestionScheme>
<d:ControlConstructScheme>
<l:LogicalProduct>
<l:DataRelationship>
<l:CategoryScheme>
<l:CodeScheme>
<l:VariableScheme>
<p:PhysicalDataProduct>
<pi:PhysicalInstance>
<a:Archive>
<a:OrganizationScheme>

- Version 1.0.0 Preparing the proposal for funding
- Version 1.1.0 Entering funding information and revising/versioning earlier content

<s:StudyUnit>
<s:Abstract>
<s:Purpose>
<r:FundingInformation>
<c:ConceptualComponents>
<cc:Concepts>
<c:Universe>
<d:DataCollection>
<d:Methodology>
<d:QuestionScheme>
<d:ControlConstructScheme>
<l:LogicalProduct>
<l:DataRelationship>
<l:CategoryScheme>
<l:CodeScheme>
<l:VariableScheme>
<p:PhysicalDataProduct>
<pi:PhysicalInstance>
<a:Archive>
<a:OrganizationScheme>

- Version 1.0.0 Preparing the proposal for funding
- Version 1.1.0 Entering funding information and revising/versioning earlier content
- Version 2.0.0 Preparing for data collection

<s:StudyUnit>
<s:Abstract>
<s:Purpose>
<r:FundingInformation>
<c:ConceptualComponents>
<cc:Concepts>
<c:Universe>
<d:DataCollection>
<d:Methodology>
<d:QuestionScheme>
<d:ControlConstructScheme>
<l:LogicalProduct>
<l:DataRelationship>
<l:CategoryScheme>
<l:CodeScheme>
<l:VariableScheme>
<p:PhysicalDataProduct>
<pi:PhysicalInstance>
<a:Archive>
<a:OrganizationScheme>

- Version 1.0.0 Preparing the proposal for funding
- Version 1.1.0 Entering funding information and revising/versioning earlier content
- Version 2.0.0 Preparing for data collection
- Version 3.0.0 Completing the study and preparing the data

Capture of Metadata Regarding Data Use

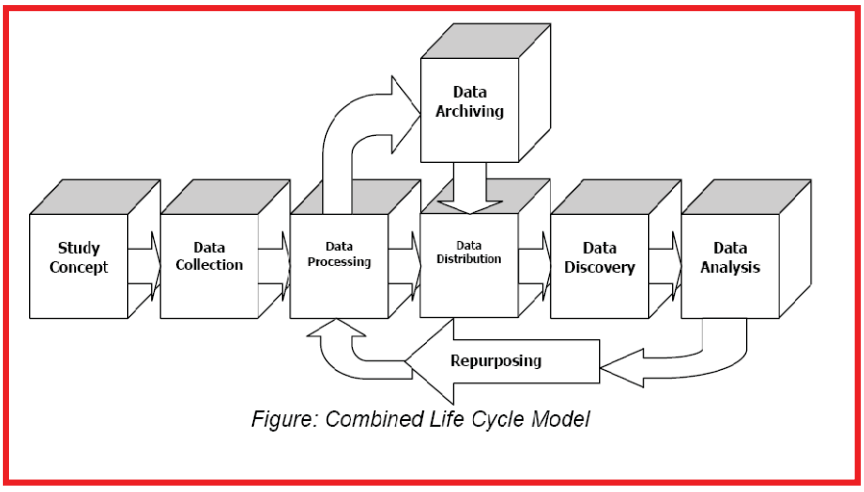
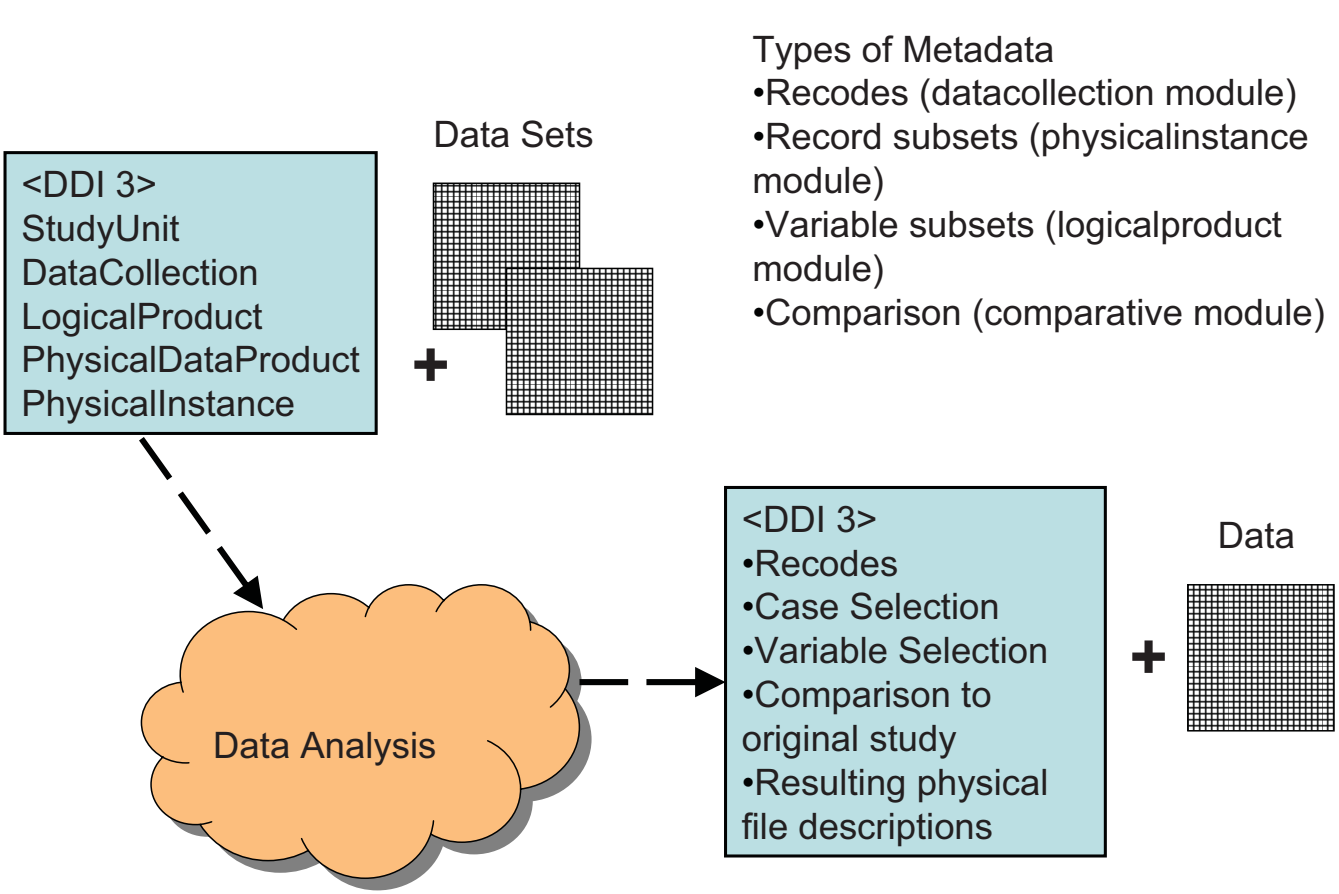


Figure: Combined Life Cycle Model

- This use case concerns how DDI 3 can capture information about how researchers use data, which can then be added to the overall metadata set about the data sources they have accessed.



Metadata Mining for Comparison, etc.

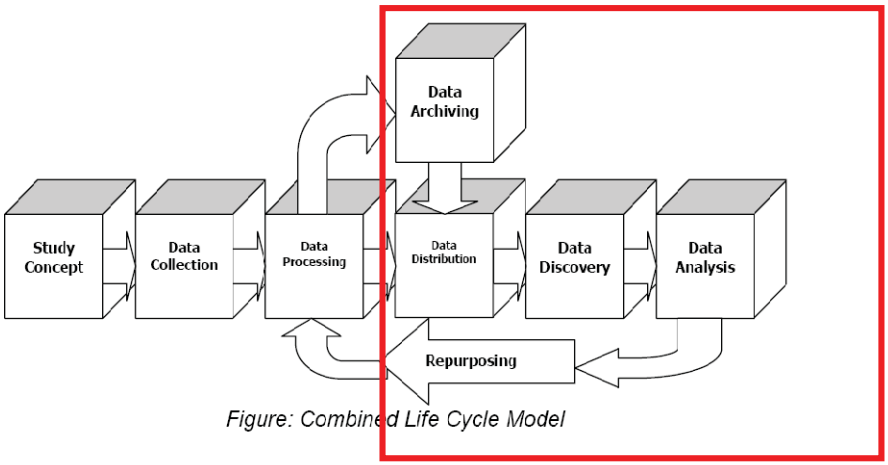
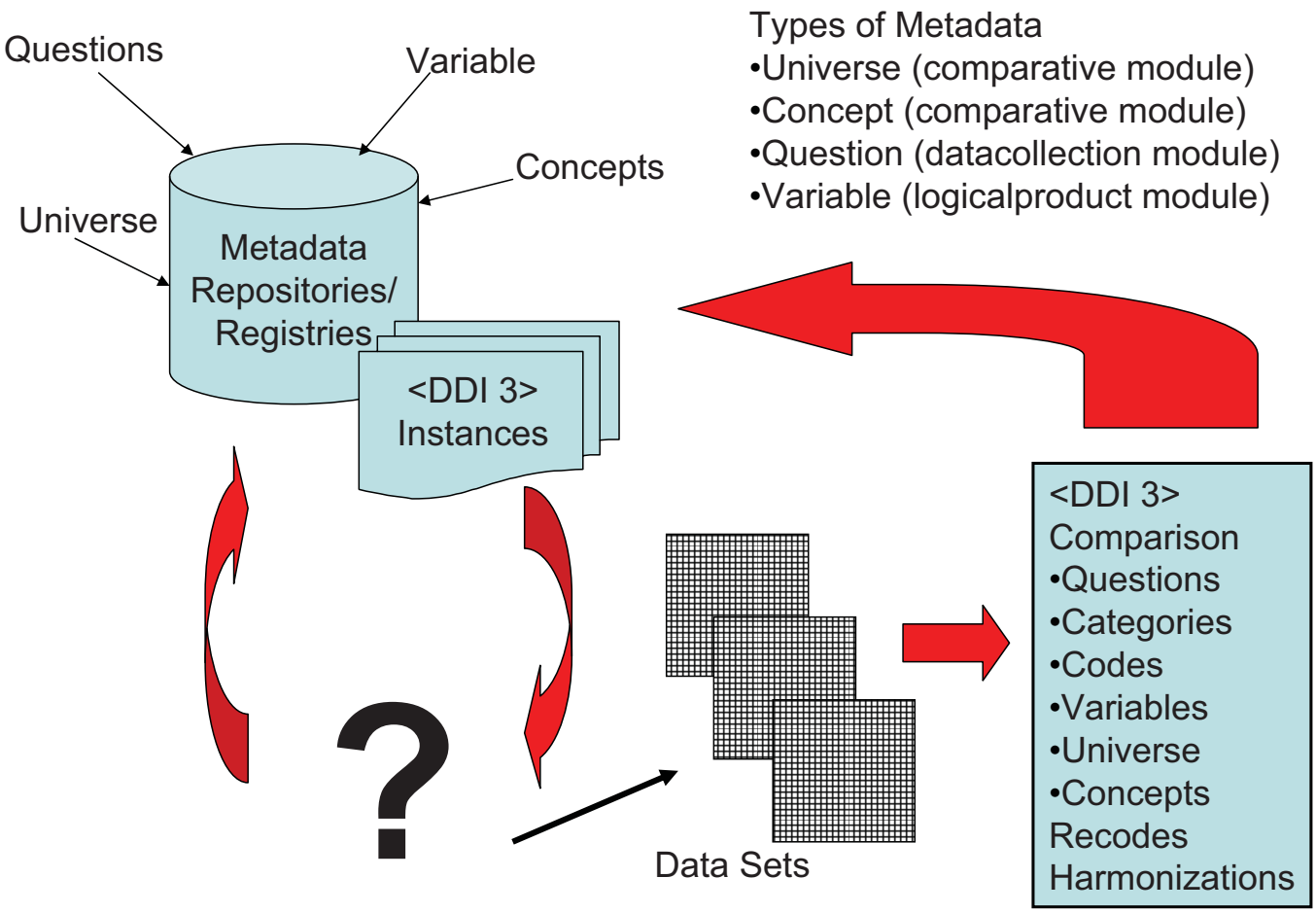


Figure: Combined Life Cycle Model

- This use case concerns how collections of DDI 3 metadata can act as a resource to be explored, providing further insight into the comparability and other features of a collection of data.



Generating Instruction Packages/Presentations

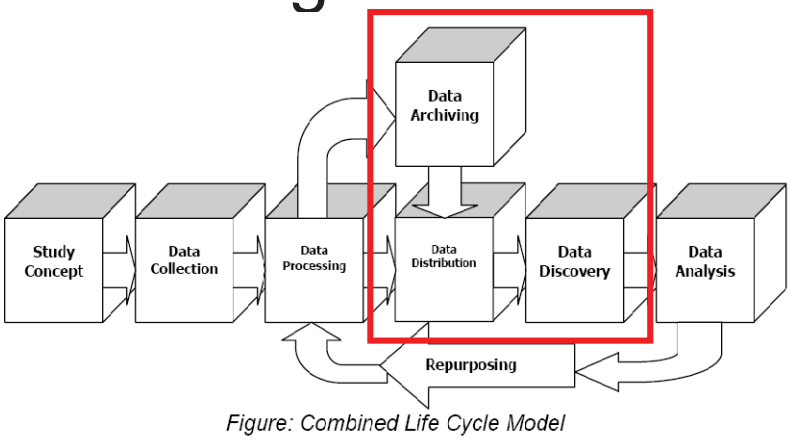
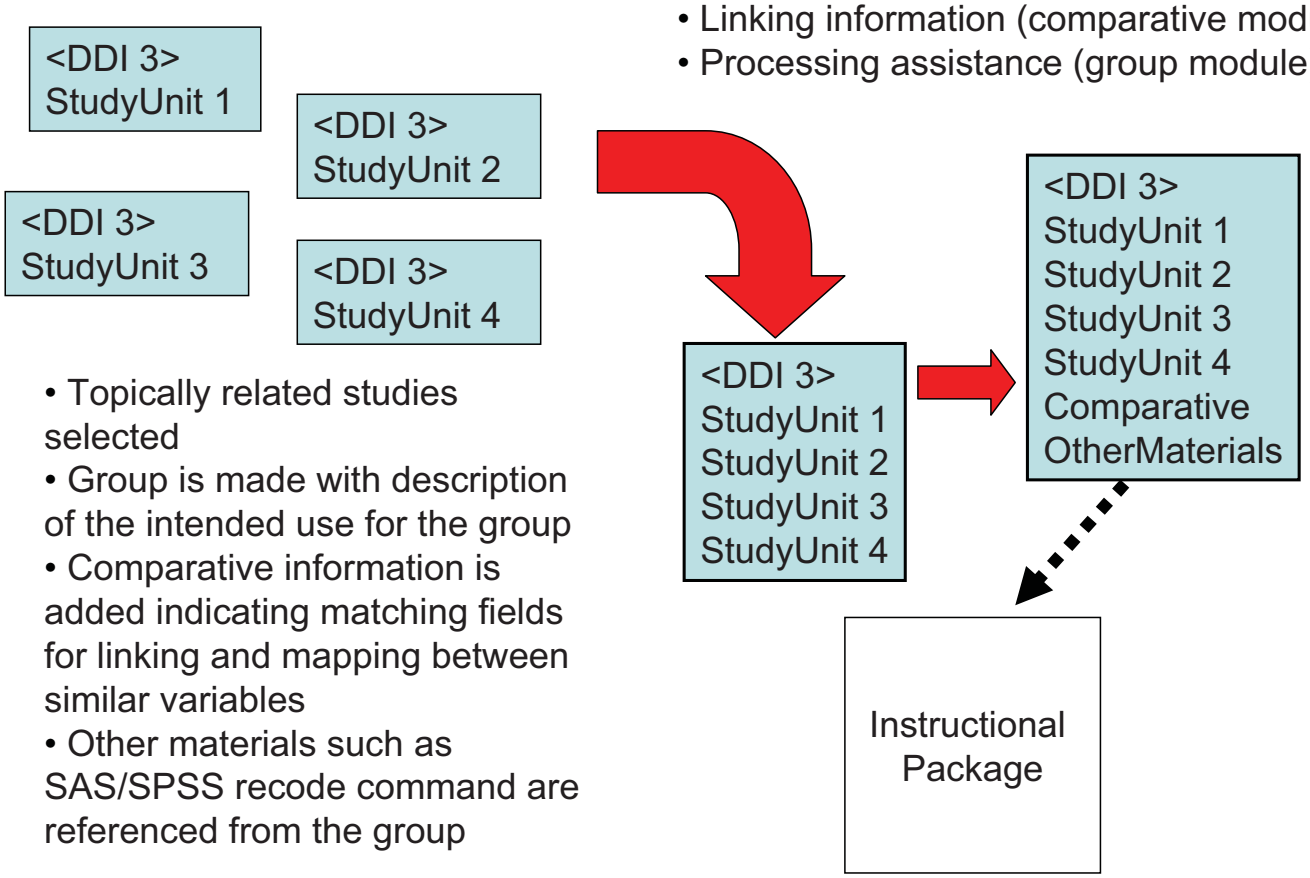


Figure: Combined Life Cycle Model

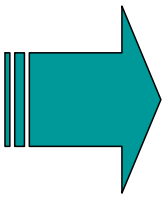
- This use case concerns how DDI 3 can support automation around the instruction of students and others.

Types of Metadata
- Individual studies (studyunit module)
- Grouping purpose (group module)
- Linking information (comparative module)
- Processing assistance (group module)



- Topically related studies selected
- Group is made with description of the intended use for the group
- Comparative information is added indicating matching fields for linking and mapping between similar variables
- Other materials such as SAS/SPSS recode command are referenced from the group

```
<s:StudyUnit
   id="SU_1">
<s:StudyUnit
   id="SU_2">
<s:StudyUnit
   id="SU_3">
<s:StudyUnit
   id="SU_4">
```



```
<g:Group>
   <g:Abstract>
   <g:Purpose>
   <g:StudyUnit>
        <g:Reference>
                <r:ID>SU_1</r:ID>
<g:StudyUnit>
        <r:Reference>
                <r:ID>SU_2</r:ID>
<g:StudyUnit>
        <r:Reference>
                <r:ID>SU_3</r:ID>
<g:StudyUnit>
        <r:Reference>
                <r:ID>SU_4</r:ID>
<cm:Comparison>
```

Register/Administrative Data

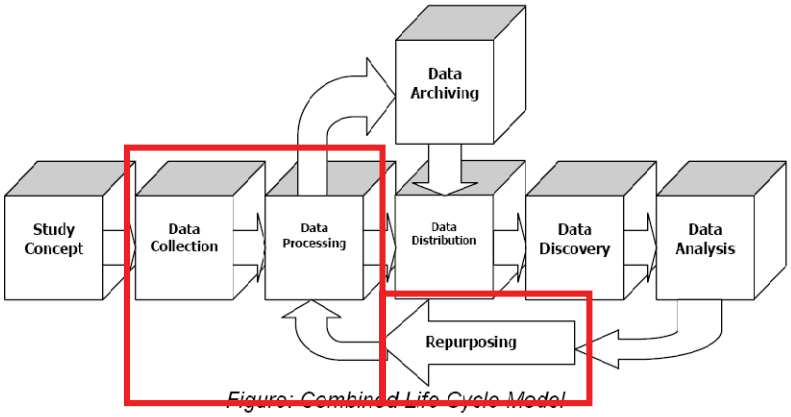
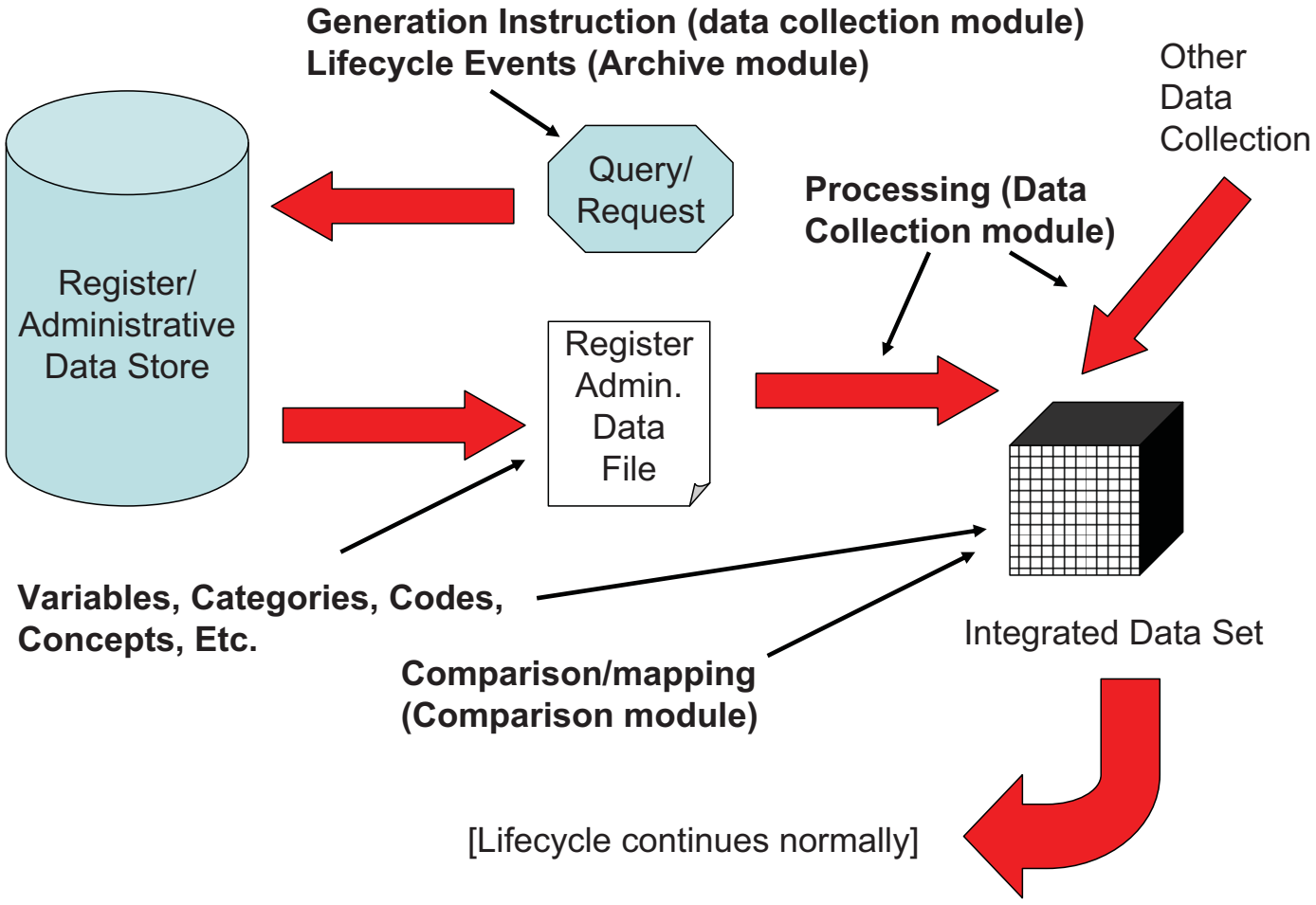


Figure: Combined Life Cycle Model

- This use case concerns how DDI 3 can support the retrieval, organization, presentation, and dissemination of register data



<g:Group>
<cm:Comparison>
<s:StudyUnitReference>
<s:StudyUnit>
<d:DataCollection>
<d:Methodology>
<d:ProcessEvent>
<l:LogicalProduct>
<l:DataRelationship>
<l:VariableScheme>
<pd:PhyscialDataProduct>
<pi:PhyscialInstance>

Emphasis is on the process of collection

May include NCube Logical Product

If data is obtained from multiple studies, Group and comparison may be used

Use Cases

- Study design/survey instrumentation
- Questionnaire generation/data collection and processing
- Data recoding, aggregation and other processing
- Data dissemination/discovery
- Archival ingestion/metadata value-add
- Question/concept/variable banks
- DDI for use within a research project
- Capture of metadata regarding data use
- Metadata mining for comparison, etc.
- Generating instruction packages/presentations
- Registry/Administrative data

Group

Grouping and Inheritance

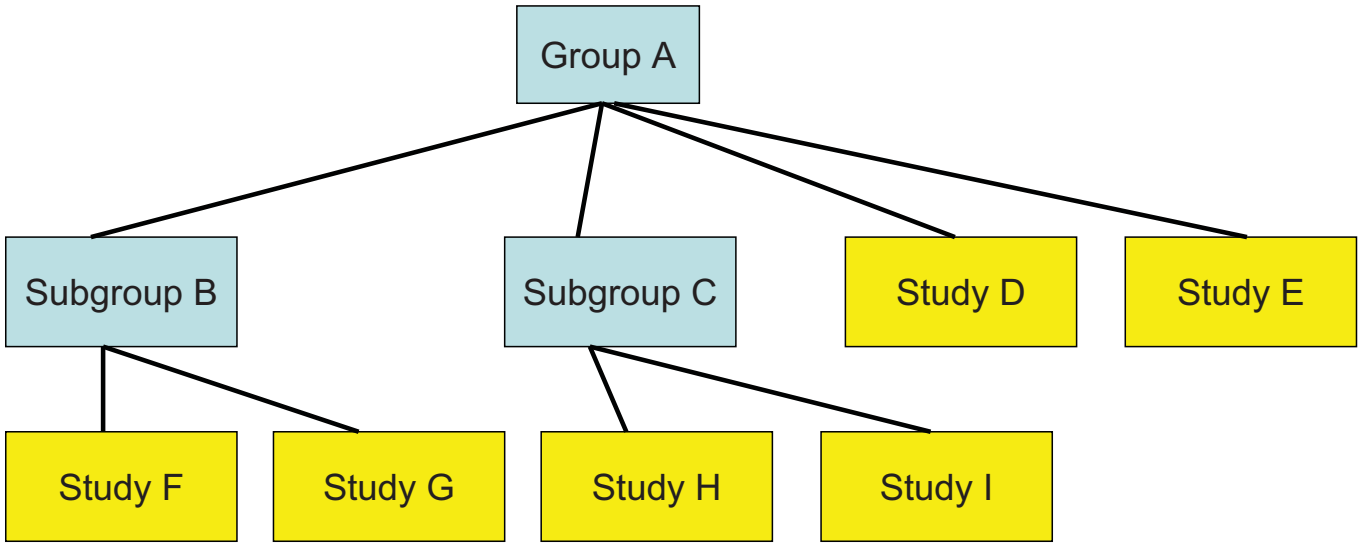
Group:Grouping and Inheritance

- Grouping is the feature which allows DDI 3 to package groups of studies into a single XML instance, and express relationships between them
- To save repetition – and promote re-use – there is an inheritance mechanism, which allows metadata to be automatically shared by studies
- This can be a complicated topic, but it is the basis for many of DDI 3.'s features, including comparison of studies
- There is a switch which can be used to "turn off" inheritance

Group Contents

- A group can contain study units, subgroups, and resource packages:
  - Study units document individual studies
  - Subgroups (inline or by reference)
  - Any of the content modules (Logical Product, Data Collection, etc.)
- Groups can nest indefinitely
- They have a set of attributes which explain the purpose of the group (as well as having a human-readable description):
  - Grouping by Time
  - Grouping by Instrument
  - Grouping by Panel
  - Grouping by Geography
  - Grouping by Data Set
  - Grouping by Language
  - Grouping by User-Defined Factor

Inheritance



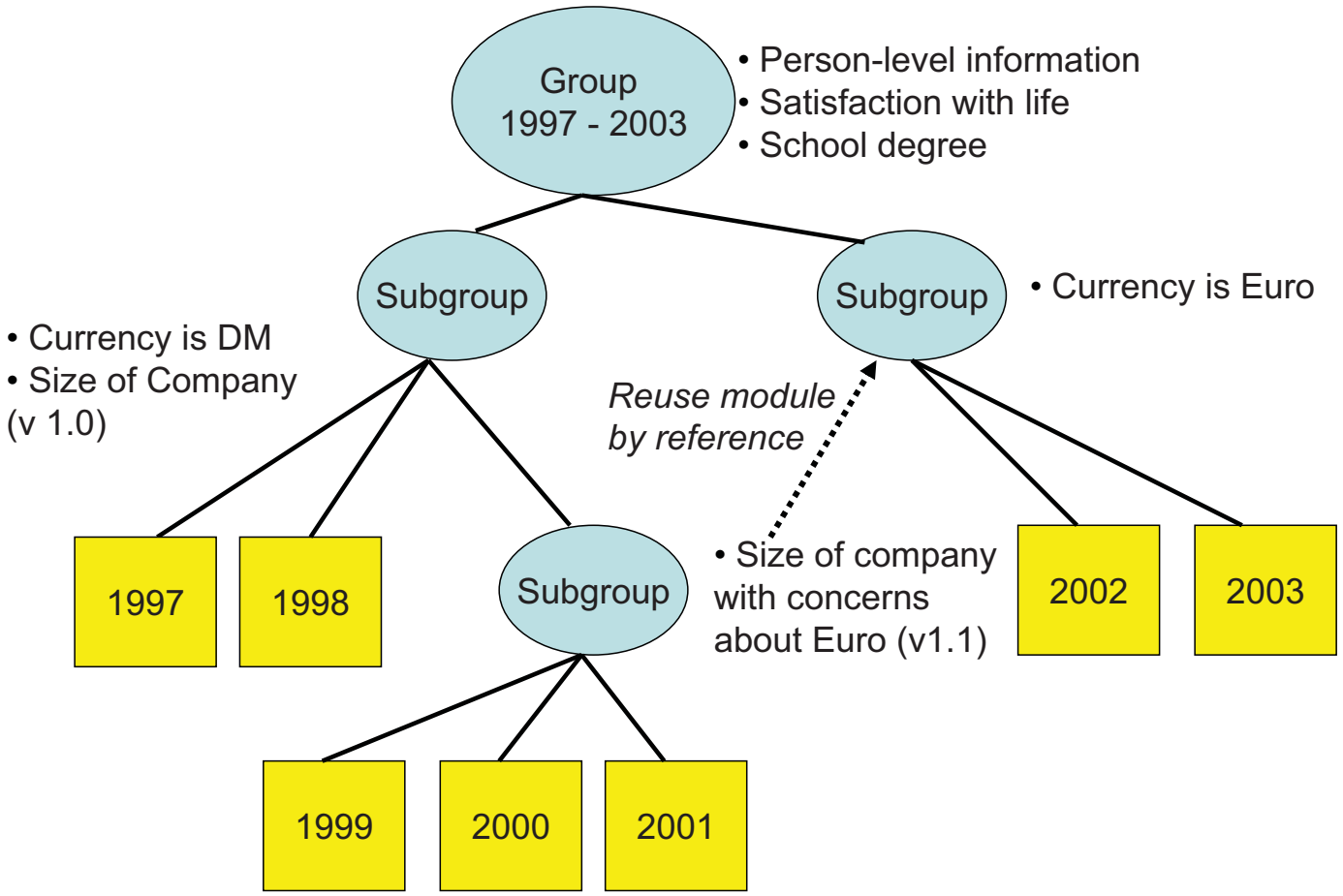
• Modules can be attached at any level
• They are shared – without repetition – by all child study units and subgroups
• If Group A has declared a concept called “X”, it is available to Study Units D – I.
• If Subgroup C has declared a Variable “Gender”, it is available to Study Units H and I without reference or repetition
• Inherited metadata can be changed using local overrides which add, replace, or delete inherited properties

Actions in Identifiers

- In some places – especially in groups where lots of metadata is being inherited – you can Add, Replace, and Delete items using identifiers.
  - Using @action attribute = Add/Delete/Replace
  - Repeat the identifier of the inherited object being locally modified
- This allows for local re-definition that is *not* reflected in a new version of the scheme
  - It cannot be reused
- For re-use, schemes should be versioned!

German Social Economic Panel (SOEP) Study Example

- The following slides show how different types of metadata can be shared using grouping and inheritance
- The SOEP is a panel study, with different panels on different years
  - Variables change over time
  - New questions and data are added


Group
1997 - 2003
• Person-level information
• Satisfaction with life
• School degree
Subgroup
Subgroup
• Currency is Euro
• Currency is DM
• Size of Company
(v 1.0)
Reuse module
by reference
1997
1998
Subgroup
• Size of company
with concerns
about Euro (v1.1)
2002
2003
1999
2000
2001

Comparison

Comparison

- There are two types of comparison in DDI 3.0:
  - Comparison by design
  - Ad-hoc (after-the-fact) comparison
- Comparison by design can be expressed using the grouping and inheritance mechanism
- Ad-hoc comparison can be described using the comparison module
- The comparison module is also useful for describing harmonization when performing case selection activities

Comparison Content

- A comparison element is placed on a group or subgroup
- It contains:
  - Description of the comparison
  - Concept maps
  - Variable maps
  - Question maps
  - Category maps
  - Code maps
  - Universe maps
  - Notes
- Each map provides for a description of how two compared items correlate and/or differ, and also allows for a coding to be associated with the correlation

Ad Hoc Groups

- Creating a course specific group
  - 3 files on aging
  - Create the group and declare the reason for selecting and including these studies
  - Note common or comparable concepts OR clarify why they are similar but NOT the same
  - Map any needed recodes for comparability
  - Provide the links (for example geographic)

Equivalencies

- FIPS
  - 01 Alabama
  - 02 Alaska
  - 04 Arkansas
  - 06 California
  - 08 Colorado
  - 09 Connecticut
  - 10 Delaware
  - 11 District of Columbia
  - 12 Florida

=

- CENSUS
  - 63 Alabama
  - 94 Alaska
  - 86 Arkansas
  - 71 California
  - 84 Colorado
  - 16 Connecticut
  - 51 Delaware
  - 53 District of Columbia
  - 59 Florida

Providing Comparative Information

- Create the category and coding schemes
- Use the comparison maps to provide comparability
  - Codes, Categories, Variables, Concepts Questions, Universe
- Example:
  - 6 files using 3 different age variables
  - Single year, five year, and ten year cohorts
- Map each equivalent structure to a single example
- Map the single year to the five year
- Map the five year to the ten year
- Provide the software command to do the conversion

SINGLE YEARS

< 1 year
1 year
2 years
3 years
4 years
5 years
6 years
7 years
8 years
9 years
10 years
11 years
12 years
13 years
14 years
15 years
16 years
17 years
18 years
19 years
20 years
Etc.

5 YEAR COHORTS

< 5 years

5 to 9 years

10 to 14 years

15 to 19 years

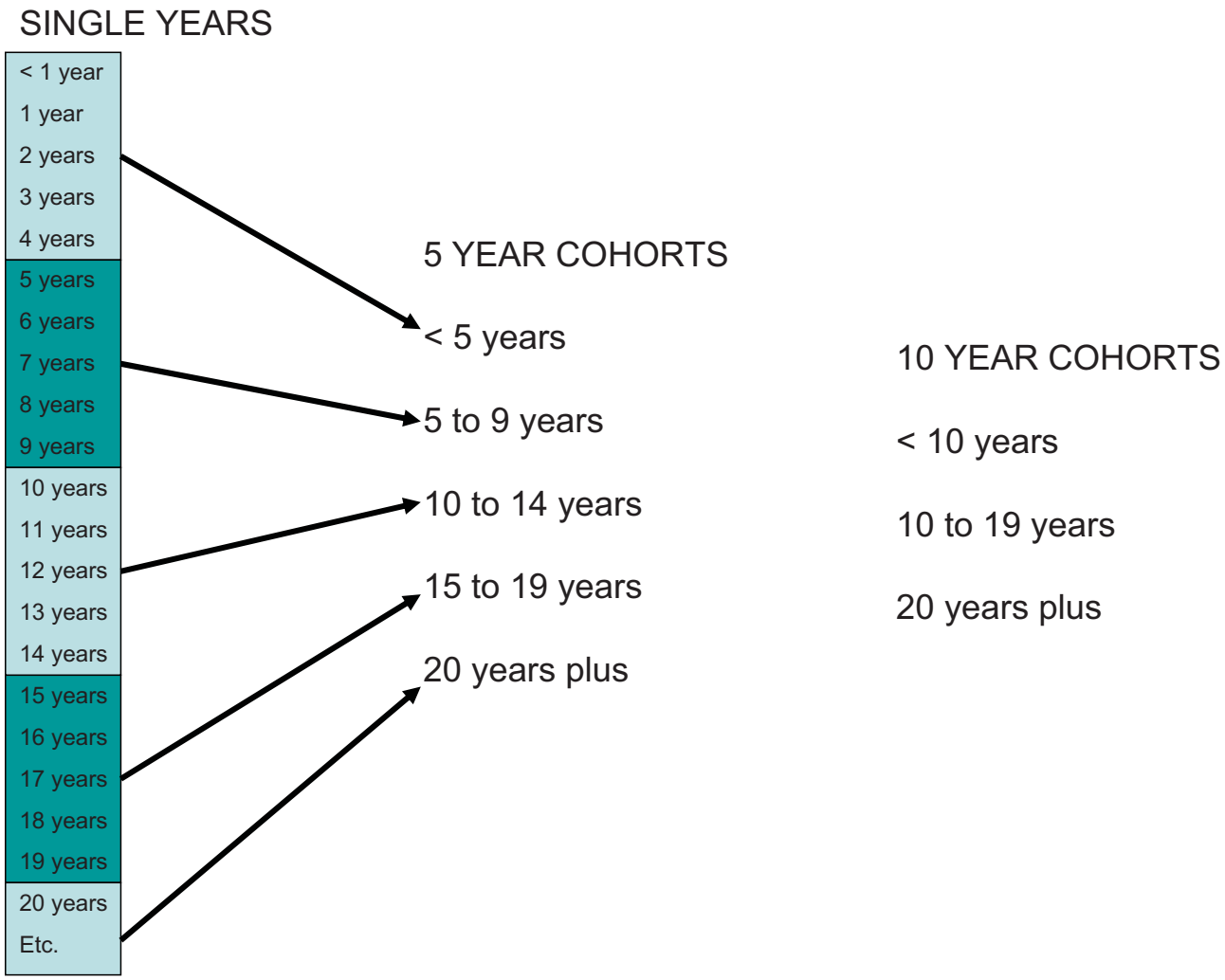
20 years plus

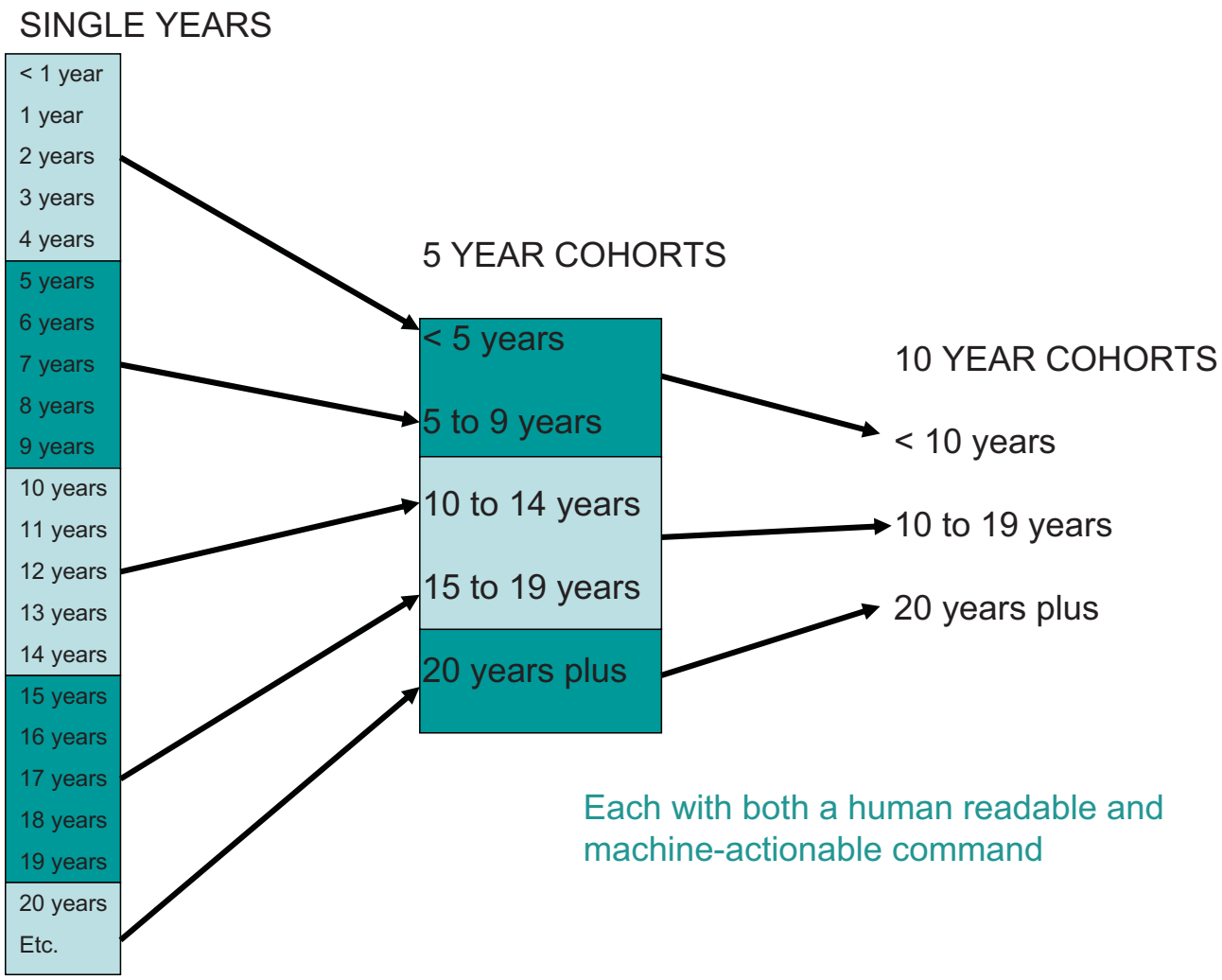
10 YEAR COHORTS

< 10 years

10 to 19 years

20 years plus


SINGLE YEARS
< 1 year
1 year
2 years
3 years
4 years
5 years
6 years
7 years
8 years
9 years
10 years
11 years
12 years
13 years
14 years
15 years
16 years
17 years
18 years
19 years
20 years
Etc.
5 YEAR COHORTS
< 5 years
5 to 9 years
10 to 14 years
15 to 19 years
20 years plus
10 YEAR COHORTS
< 10 years
10 to 19 years
20 years plus


SINGLE YEARS
< 1 year
1 year
2 years
3 years
4 years
5 years
6 years
7 years
8 years
9 years
10 years
11 years
12 years
13 years
14 years
15 years
16 years
17 years
18 years
19 years
20 years
Etc.
5 YEAR COHORTS
< 5 years
5 to 9 years
10 to 14 years
15 to 19 years
20 years plus
10 YEAR COHORTS
< 10 years
10 to 19 years
20 years plus
Each with both a human readable and machine-actionable command

Detail of question comparability

Comparison Map	Textual Content of Main Body		Category		Code Scheme	
	Same	*Similar*	*Same*	*Similar*	*Same*	*Different*
Question	X		X		X	
	X		X			X
	X			X	X	
	X			X		X
		X	X		X	
		X	X			X
		X		X	X	
		X		X		X

Relationship of DDI to other standard

Relationship to Other Standards: Archival

- Dublin Core
  - Basic bibliographic citation information
  - Basic holdings and format information
- METS
  - Upper level descriptive information for managing digital objects
  - Provides specified structures for domain specific metadata
- OAIS
  - Reference model for the archival lifecycle
- PREMIS
  - Supports and documents the digital preservation process

Dublin Core [AGLS]

- Purpose: describe resources
  - Standard for cross-domain information resource description
  - Widely used to describe digital materials such as video, sound, image, text, and composite media
  - Small core set of elements – can be extended
  - Used for survey documentation
- Sponsors: Dublin Core Metadata Initiative
- http://dublincore.org/

METS: Metadata Encoding & Transmission Standard

- A standard for encoding descriptive, administrative, and structural metadata regarding objects within a digital library
- Expressed using XML Schema
- Maintained in the Network Development and MARC Standards Office of the Library of Congress,
- Developed as an initiative of the Digital Library Federation.
- The editorial board endorses DDI for use with METS
- http://www.loc.gov/standards/mets/

OAIS: Open Archival Information System

- Addresses a full range of archival information preservation functions including ingest, archival storage, data management, access, and dissemination.
- ISO 14721:2003
- http://nost.gsfc.nasa.gov/isoas/

PREMIS: Preservation Metadata Implementation Strategies

- Preservation metadata makes digital objects self documenting over time
- XML based standard which can be used as an implementation of OAIS or other archival model
- Addresses:
  - Provenance
  - Authenticity
  - Preservation activity
  - Technical environment
  - Rights management
- http://www.loc.gov/standards/premis/

Relationship to Other Standards: Non-Archival

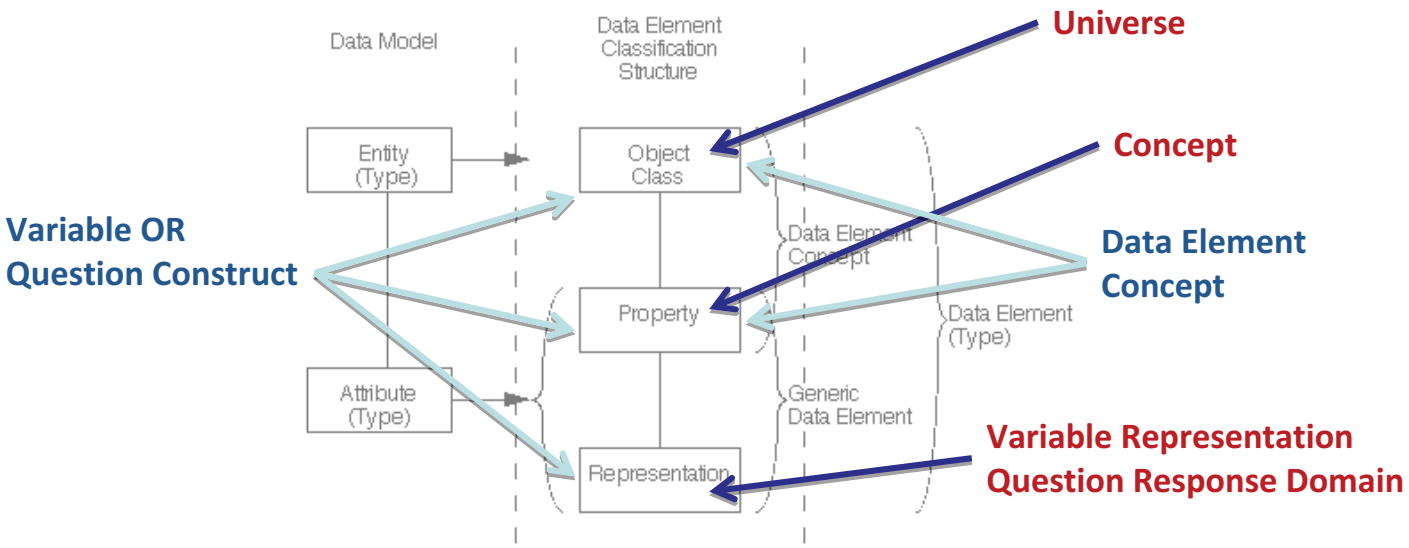
- ISO 19115 – Geography
  - Metadata structure for describing geographic feature files such as shape, boundary, or map image files and their associated attributes
- ISO/IEC 11179
  - International standard for representing metadata in a Metadata Registry
  - Consists of a hierarchy of “concepts” with associated properties for each concept
- SDMX
  - Exchange of statistical information (time series/indicators)
  - Supports metadata capture as well as implementation of registries

ISO 19115

- Purpose: Capture geography
  - It is a component of the series of ISO 191xx standards for Geospatial metadata.
  - ISO 19115 defines how to describe geographical information and associated services, including contents, spatial-temporal purchases, data quality, access and rights to use.
  - Compliance in DDI 3
- Sponsors: ISO/TC 211 Geographic information/Geomatics
- http://www.isotc211.org/

ISO/IEC 11179

- Purpose: Manage registries / concepts
  - International standard for representing metadata for an organization in a Metadata Registry (a central location in an organization where metadata definitions are stored and maintained in a controlled fashion)
  - Compliance with this standard is important for other standards, and both DDI 3 and SDMX have mapping mechanisms
- Sponsors: ISO/IEC Joint Technical Committee on Metadata Standards
- http://metadata-standards.org/



ISO/IEC 11179-1

International Standard ISO/IEC 11179-1: Information technology – Specification and standardization of data elements – Part 1: Framework for the specification and standardization of data elements Technologies de l'informatin – Spécifiction et normalization des elements de données – Partie 1: Cadre pout la specification et la normalization des elements de données. First edition 1999-12-01 (p26) http://metadata-standards.org/11179-1/ISO-IEC_11179-1_1999_IS_E.pdf

Statistical Data and Metadata Exchange (SDMX)

- Purpose: Exchange of statistical information (time series/indicators).
  - Covers the metadata capture as well as implementation of registries.
  - Currently version 2.0 and also an ISO standard (17369:2005)
- Sponsors: Bank for International Settlements (BIS), European Central Bank (ECB), EUROSTAT, International Monetary Fund (IMF), Organization for Economic Cooperation and Development (OECD), United Nations (UN), World Bank
- Can actually be used for many other purposes. It's a metadata metadata model.
- http://www.sdmx.org

DDI 3 & SDMX 2.0

- Are complementary specifications
- DDI 3 and SDMX 2.0 have been designed to work with each other
  - SDMX registries can wrap DDI documents
  - Microdata: single point in time / geography, high level of details (for statisticians, researchers)
  - Macrodata: high level indicators across time and geography (fro economists, policy makers)
  - Using DDI+SDMX allows linkages and drilling down from indicator to its source

- See "DDI and SDMX: Complementary, Not Competing, Standards", A. Gregory, P. Heus, July 2007 available at http://www.opendatafoundation.org/?lvl1=resources&lvl2=papers

Some major XML metadata specifications for data content management

- Statistical Data and Metadata Exchange (SDMX)
  - Macrodata, time series, indicators, registries
  - http://www.sdmx.org
- **Data Documentation Initiative (DDI)**
  - **Microdata (surveys, studies), aggregate, administrative data**
  - **http://www.ddialliance.org**
- ISO/IEC 11179
  - Semantic modeling, concepts, registries
  - http://metadata-standards.org/11179/
- ISO 19115
  - Geography
  - http://www.isotc211.org/
- Dublin Core
  - General resources (documentation, images, multimedia)
  - http://www.dublincore.org

Interacting Standards for Data

- Dublin Core
- ISO/IEC 11179
- ISO 19115 – Geography
- Statistical Packages
- METS
- PREMIS
- SDMX
- DDI

- Citation structure
- Coverage
  - Temporal
  - Topical
  - Spatial
- Location specific information

Interacting Standards for Data

- Dublin Core
- ISO/IEC 11179
- ISO 19115 – Geography
- Statistical Packages
- METS
- PREMIS
- SDMX
- DDI

- Structure and content of a data element as the building block of information
- Supports registry functions
- Provides
  - Object
  - Property
  - Representation

Interacting Standards for Data

- Dublin Core
- ISO/IEC 11179
- ISO 19115 – Geography
- Statistical Packages
- METS
- PREMIS
- SDMX
- DDI

- i.e., ANZLIC and US FGDC
- Focus is on describing spatial objects and their attributes

Interacting Standards for Data

- Dublin Core
- ISO/IEC 11179
- ISO 19115 – Geography
- Statistical Packages
- METS
- PREMIS
- SDMX
- DDI

- Proprietary standards
- Content is generally limited to:
  - Variable name
  - Variable label
  - Data type and structure
  - Category labels
- Translation tools used to transport content

Interacting Standards for Data

- Dublin Core
- ISO/IEC 11179
- ISO 19115 – Geography
- Statistical Packages
- METS
- PREMIS
- SDMX
- DDI

- Digital Library Federation
- Consistent outer wrapper for digital objects of all type
- Contains a profile providing the structural information for the contained object

Interacting Standards for Data

- Dublin Core
- ISO/IEC 11179
- ISO 19115 – Geography
- Statistical Packages
- METS
- PREMIS
- SDMX
- DDI

- Preservation information for digital objects

Interacting Standards for Data

- Dublin Core
- ISO/IEC 11179
- ISO 19115 – Geography
- Statistical Packages
- METS
- PREMIS
- SDMX
- DDI

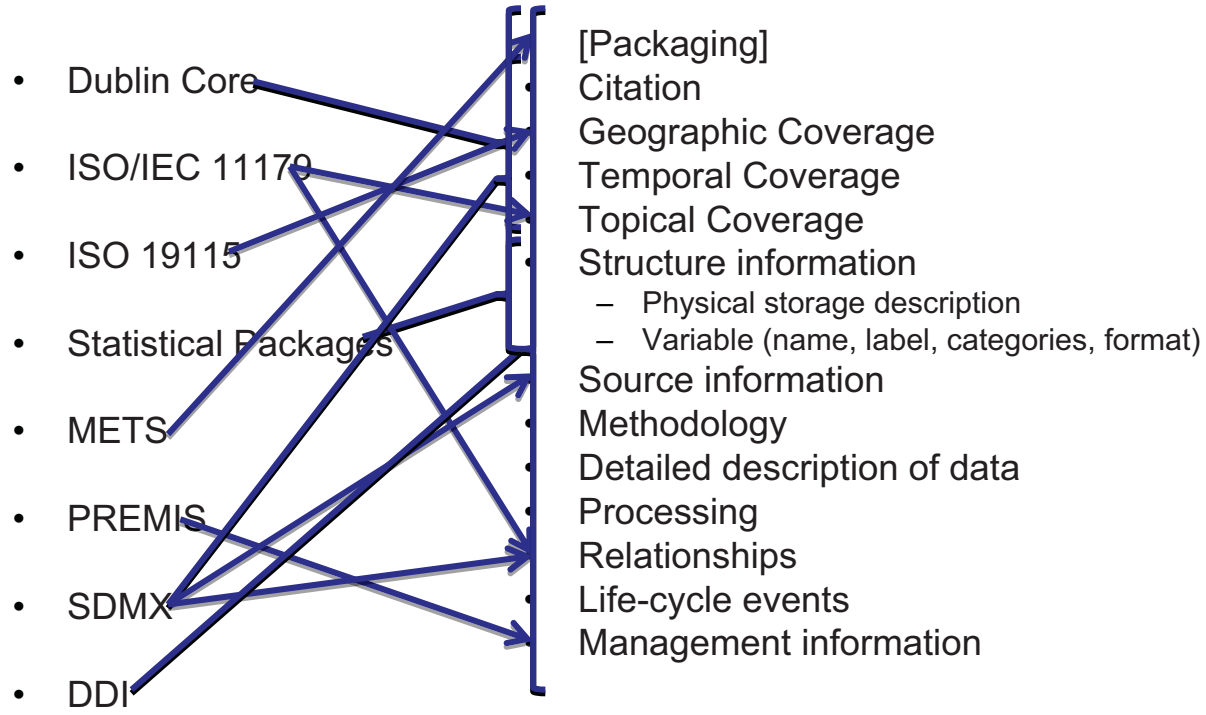
- Developed for statistical tables
- Supports well structured, well defined data, particularly time-series data
- Contains both metadata and data
- Supports transfer of data between systems

Interacting Standards for Data

- Dublin Core
- ISO/IEC 11179
- ISO 19115 – Geography
- Statistical Packages
- METS
- PREMIS
- SDMX
- DDI

- Version 3.0 covers life-cycle of data and metadata
- Data collection
- Processing
- Management
- Reuse or repurposing
- Support for registries
- Grouping & Comparison

Metadata Coverage



DDI and SDMX Combined Data Model

- DDI 3 focuses on:
  - collection and production of microdata
  - reuse and sharing of common data structures
  - conversion to statistical tables (matrices)
  - preservation and multiple storage options
- SDMX focuses on:
  - statistical tables
  - reuse and sharing of common data structures
  - consistant data transfer structure
- Together they form a coherent data management model for data capture, storage and interchange with a wide area of overlap

Future Developments

New Features

- Currently there are three active committees working on new features
  - Qualitative Data
  - Survey Design & Implementation
  - Controlled Vocabulary
- Many other topics have been mentioned, but there are not currently active committees

Survey Design & Implementation

- Covers the process prior to data collection in the lifecycle
  - Survey methodology (sampling)
  - Questionnaire development (question testing, order testing, comparable language, etc.)
  - Reviewing the concept structures

Qualitative Data

- This group's work is based on a prototype developed at UKDA – QuDEX - working with the vendors of Computer-Assisted Qualitative Data Analysis Software (CAQDAS)
- Useful for adding metadata and annotations to textual information (interviews, etc.)
- Includes all types of media (including mixed media)

Controlled Vocabularies

- Provision of a base set of controlled vocabularies for community use
- Currently under content review
- Rules and procedures for using Genericode to publish controlled vocabularies

Future Directions

- Preservation
  - Closer alignment with OAIS features
  - Improved interaction with PREMIS Quality metadata
- Quality assessment in terms of both process and data evaluation
  - This may be based on national and international data quality frameworks such as DQAF (from the IMF) and similar European frameworks (Eurostat, OECD)
- Review for coverage of Register / Administrative data, may result in additional metadata capture

New Features?

- Alliance members can suggest new features
  - A call for participation will go out
  - Prioritization is driven by the membership
  - If there is interest, the work can be started
- Existing prior work is generally taken as a starting point, but will not be adopted wholesale
- Joining the DDI Alliance will provide input into this process
  - It is generally open and receptive

Known Issues

- DDI tracks all issues in a bug-tracking system: http://mantis.ddialliance.org/
- Anyone can log in as a guest and view the issues. Click on “View Issues” to see full list
- Some aspects of the DDI 3 design were put off for future versions, and will be addressed moving forward

Process

- The process for making fixes to the XML schemas and other work will be:
  - TIC will review bugs as they are reported, and recommend to the Director if a new release is needed
    - New releases will typically be very minor (documentation changes, minor bug-fixes)
  - It is possible to have a new release as often as every 3 months
    - This is very unlikely!

Strategic Plan for DDI

- Can be found at: http://www.ddialliance.org/DDI/org/strategic-plan.pdf
- Describes the future directions and strategy of DDI for those who are thinking of joining the organization

DDI Resources

- DDI Alliance Site
  - http://www.ddialliance.org
  - General link to all resources/news
  - Link to Sourceforge for standards distributions
  - Link to prototype page – good for examples
- Tools/Resources Page
  - http://tools.ddialliance.org
  - Best place for tools, slides, and resources

DDI Resources (cont.)

- Mailing Lists
  - www.icpsr.umich.edu/mailman/admin/
  - All of the lists starting with "DDI" are related to DDI topics
    - General list
    - List for each sub-committee
    - Not all groups are active
    - User list is the best general place
- Open Data Foundation Site
  - www.opendatafoundation.org
  - White papers, other resources/tools

DDI Resources (cont.)

- DDI Agency Registry
  - http://tools.ddialliance.org/?lvl1=community&lvl2=agencyid
  - Sign up for unique global agency identifier – helps provide interoperability between organizations
  - Currently taking pre-registrations – site will be permanent in future
- International Household Survey Network
  - http://surveynetwork.org
  - DDI 2.*-based toolkit available for developing countries (some free tools)
  - Catalog of surveys, many documented in DDI

Best Practices

- Governance
- Workflows
- Versioning and Publication
- Resources Packages and Schemes
- DDI as content for registries
- URN resolution
- Management of DDI Identifiers

DDI Events

- IASSIST
  - www.iassistdata.org
  - Not an official DDI event, but many DDI-related presentations and meetings
  - DDI Alliance Expert Committee meets before or after every year
  - 36th Meeting in Ithaca NY, 1-4 June 2010
  - DDI and other Workshops given day before the meeting
  - Annual meetings go US-Canada-US-Outside North America-US-Canada-US-Outside North America etc.

DDI Events (cont.)

- European DDI User's Group
  - 2st meeting is being planned
- GESIS-Sponsored Autumn Events
  - Schloss Dagstuhl training and other meetings ('Longitudinal Survey' workshop second week!)
  - This is the 4nd year
- Open Data Foundation meetings
  - Spring meeting in Europe
  - Winter meeting in the US
  - DDI is a major topic of discussion

Tomorrow's Presentations

- What parts of the life-cycle they are collecting metadata from?
- What parts or features of DDI are they using?
- What are their goals?
- How are they storing DDI content?
  - Native XML, Data Base?
- What other standards are they working with?
- Are there other approaches?