THE “METADATA RECORD” AND DCMI ABSTRACT MODEL
<table>
<thead>
<tr>
<th>Property URI</th>
<th>Value URI</th>
</tr>
</thead>
<tbody>
<tr>
<td>dcam:memberOf</td>
<td>VES URI</td>
</tr>
<tr>
<td>rdf:value</td>
<td>^^^SES URI</td>
</tr>
</tbody>
</table>
DCAM Description Set Model

• The structure of “DC metadata”
• Uses URIs to refer to resources described & to metadata terms

• a description set is made up of one or more descriptions, each of which describes one resource
• a description is made up of
  – zero or one described resource URI
    • identifies described resource
  – one or more statements
• a statement is made up of
  – exactly one property URI
    • identifies property
  – exactly one value surrogate
• a value surrogate is either a literal value surrogate or a non-literal value surrogate

• e.g. http://dublincore.org/documents/2007/06/04/abstract-model/
• e.g. http://purl.org/dc/terms/subject
DCAM Description Set Model

- **a literal value surrogate** is made up of
  - exactly one value string
    - **encodes** value
- **a non-literal value surrogate** is made up of
  - zero or one value URIs
    - **identifies** value
  - zero or one vocabulary encoding scheme URI
    - **identifies** a set of which the value is a member
  - zero or more value strings
    - **represents** value
- **a value string** is either a **plain value string** or a **typed value string**
  - a plain value string may have an associated value string language
  - a typed value string is associated with a syntax encoding scheme URI
- **Vocabulary Encoding Scheme**
  - A named set to which a “Thing” belongs
- **Syntax Encoding Scheme**
  - A named set of rules for the “interpretation” of a set of “Strings”
Example: Description of document, description of publisher

Description

Statement

<http://purl.org/dc/terms/publisher>

<http://example.org/org/DCMI>

Non-Literal Value Surrogate

<http://purl.org/dc/terms/subject>

<http://example.org/mySH/h123>

<http://example.org/terms/mySH>

“Metadata” en

"Métadonnées" fr

Description

<http://example.org/org/DCMI>

Statement

<http://xmlns.com/foaf/0.1/name>

“Dublin Core Metadata Initiative” en

Literal Value Surrogate

<http://purl.org/dc/terms/created>

“1995” ^^xsd:gyear

Literal Value Surrogate
DCAM Description Set Model
ENCODING GUIDELINES BASED ON DCMI ABSTRACT MODEL
DC-RDF

• Any concrete syntax for RDF can be used for Dublin Core metadata
  – Available syntaxes include RDF/XML, N3, Turtle, RDFa
• “Expressing DC metadata using RDF” (2008)
  – http://dublincore.org/documents/2008/01/14/dc-rdf/
  – Uses RDF abstract syntax
  – Supports full DCAM description model
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dcterms="http://purl.org/dc/terms/"
  xmlns:dcam="http://purl.org/dc/dcam/
  xmlns:foaf="http://xmlns.com/foaf/0.1/">
  <rdf:Description rdf:about="http://dublincore.org/documents/2007/06/04/abstract-model/">
    <dcterms:publisher rdf:resource="http://example.org/org/DCMI" />
    <dcterms:subject>
      <rdf:Description rdf:about="http://example.org/mySH/h123">
        <dcam:memberOf rdf:resource="http://example.org/terms/mySH" />
        <rdf:value xml:lang="en">Metadata</rdf:value>
        <rdf:value xml:lang="fr">Métadonnées</rdf:value>
      </rdf:Description>
    </dcterms:subject>
  </rdf:Description>
  <rdf:Description rdf:about="http://example.org/org/DCMI">
    <foaf:name xml:lang="en">Dublin Core Metadata Initiative</foaf:name>
    <dcterms:created rdf:datatype="http://www.w3.org/2001/XMLSchema#gyear">1995</dcterms:created>
  </rdf:Description>
</rdf:RDF>
DC-Text

• “Expressing DC metadata using DC-Text”, DCMI Recommended Resource, 2007-12-03
  – Supports full DCAM description model
  – Intended for human-readability rather than machine-processing
@prefix page: <http://dublincore.org/pages/> .
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix externs: <http://example.org/terms/> .
@prefix exsh: <http://example.org/sh/> .

DescriptionSet ( 
  Description ( 
    ResourceURI ( page:home ) 
    Statement ( 
      PropertyURI ( dcterms:subject ) 
      ValueURI ( exsh:metadata ) 
      VocabularyEncodingSchemeURI ( externs:EXSH ) 
      ValueString ( "Metadata" 
        Language ( en )
      )
    )
    ValueString ( "Métadonnées" 
      Language ( fr )
    )
  )
)
)
DC-HTML

• RDFa is an excellent option for embedding Dublin Core descriptions in HTML documents
• “Expressing DC metadata using HTML/XHTML meta and link elements”
  – Supports subset of DCAM description model
  – DC metadata in HTML document describes that document
    • or at least document of which HTML page is representation
  – An HTML meta-data profile
  – GRDDL Profile Transformation to generate RDF/XML
Dublin Core metadata in XML

• “Expressing DC Description Sets using XML (DC-DS-XML)”
  – Supports full DCAM description model
  –Verbose, but easily processable
  – GRDDL Namespace Transformation to generate RDF/XML
  <dcds:description dcds:resourceURI="http://dublincore.org/pages/home">
    <dcds:statement dcds:propertyURI="http://purl.org/dc/terms/publisher"
        dcds:valueURI="http://example.org/org/DCMI"/>
    <dcds:statement dcds:propertyURI="http://purl.org/dc/terms/subject"
        dcds:vesURI="http://example.org/terms/mySH"
        dcds:valueURI="http://example.org/mySH/h123">
      <dcds:valueString xml:lang="en">Metadata</dcds:valueString>
      <dcds:valueString xml:lang="fr">Métadonnées</dcds:valueString>
    </dcds:statement>
  </dcds:description>

  <dcds:description dcds:resourceURI="http://example.org/org/DCMI">
    <dcds:statement dcds:propertyURI="http://xmlns.com/foaf/0.1/name">
      <dcds:literalValueString xml:lang="en">Dublin Core Metadata Initiative</dcds:literalValueString>
    </dcds:statement>
    <dcds:statement dcds:propertyURI="http://purl.org/dc/terms/created">
    </dcds:statement>
  </dcds:description>
</dcds:descriptionSet>
Metadata Records and RDF

- Dublin Core terms can be used in RDF without using DC-AM
- In using the DC-AM, you are also using RDF
  - DC-AM adds the notion of a bounded “record” (description set)
- Other RDF-based “record” formats under development using SPARQL Named Graphs
SINGAPORE FRAMEWORK
Dublin Core Application Profile

• Specification of how to construct description sets (descriptions, statements) to serve some purpose

• At core, a profile of a “description set”
  – a set of constraints on the description set
  – based on E-R model of problem space

• Enables
  – structural validation
  – predictability for processing, querying etc
Singapore Framework

Application Profile

Foundation standards

Domain standards
Getting from Requirements...
...to a Data Format
...based on Community Standards
...on the basis of Foundation Standards (RDF)
What does your application need to do?
Functional Requirements

• What activities must the application support?
  – Who are the users? How expert are they?
• Scholarly Works Application Profile examples:
  – “Facilitate identification of open access materials.”
  – “Be compatible with preservation metadata approaches.”
  – “Enable identification of research funder and project code.”
  – “Support navigation between different 'versions' of the same eprint.”
• DC Collections Application Profile examples:
  – “Enable searching on the entity that owns the collection”
  – “Enable selection of a collection based on a textual description”
What things are being described?
The simplest domain model...!

Resource
A slightly more complicated domain model
Domain Model for Scholarly Works Application Profile

AffiliatedInstitution

- isSupervisedBy
- isFundedBy

ScholarlyWork

- isCreatedBy
- isExpressedBy

Expression

- isManifestedBy

Manifestation

- isPublishedBy
- isAvailableAs

Agent

- isEditedBy

Copy
Description Set Profile, based on DCMI Abstract Model
DCMI Description Set Profile (DSP)

- A way of describing **structural constraints** on a description set
  - the resources that may be described by descriptions in the description set
  - the properties that may be referenced in statements
  - the ways a value surrogate may be given
- Description templates, statement templates
- Model & XML Syntax for DSP
  - Working draft by Mikael Nilsson (Royal Institute of Technology, Sweden)
Description Set Profile

<http://dublincore.org/documents/2007/06/04/abstract-model/>

Statement

<http://purl.org/dc/terms/publisher>

<http://example.org/org/DCMI>

Non-Literal Value Surrogate

<http://purl.org/dc/terms/subject>

<http://example.org/mySH/h123>

<http://example.org/terms/mySH>

"Metadata" en

"Métadonnées" fr

Value URI

Vocab Enc Scheme URI

Value String

Property URI

Description

<http://example.org/org/DCMI>

Statement

<http://xmlns.com/foaf/0.1/name>

"Dublin Core Metadata Initiative" en

Literal Value Surrogate

<http://purl.org/dc/terms/created>

"1995" ^xsd:gyear

Literal Value Surrogate

Property URI

Property URI
Example: Description Set Profile of a Book

• A book:
  – a literal title
  – a creator, described separately

• A creator
  – a literal name
Templates and constraints

Statement template: literal title

Property: dcterms:title
Literal value  Language  SES

Statement template: creator

Property: dcterms:creator  Description reference: Creator
Value URI
Vocabulary Encoding Scheme
Value string  Language  SES

Statement template: literal name

Property: foaf:name  standalone:no
Literal value  Language  SES
Documentation of templates and constraints

## Description of the eprint as a ScholarlyWork

### Entity type

<table>
<thead>
<tr>
<th>Property</th>
<th>Value URI Constraint:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min occurrence</td>
<td>1</td>
</tr>
<tr>
<td>Max occurrence</td>
<td>1</td>
</tr>
<tr>
<td>Literal?</td>
<td>No</td>
</tr>
</tbody>
</table>

**Definition**: The type nature or genre of the content of the resource.

**Eprint-specific recommendation**: Each of the entity *descriptions* in the *description sets* conforming with this application profile will need to be explicitly typed. This is done using a dc:type statement with one of the following value URIs taken from the Eprints EntityType Vocabulary Encoding Scheme corresponding to the entity being described.

### Value (Non-Literal)

<table>
<thead>
<tr>
<th>Occurrence</th>
<th>mandatory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Choose from</td>
<td><a href="http://purl.org/eprint/entityType/ScholarlyWork/">http://purl.org/eprint/entityType/ScholarlyWork/</a></td>
</tr>
</tbody>
</table>

**Vocabulary Encoding Scheme Constraint**

<table>
<thead>
<tr>
<th>Occurrence</th>
<th>mandatory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Choose from</td>
<td><a href="http://purl.org/eprint/entityType/">http://purl.org/eprint/entityType/</a></td>
</tr>
</tbody>
</table>

**Value String Constraint**: Max occurrence 0

**For example:**

```java
Statement {
    Property URI ( dc:type )
    Vocabulary Encoding Scheme URI ( eprint:EntityTypes )
    Value URI ( <http://purl.org/eprint/entityType/ScholarlyWork> )
}
```
Metadata vocabularies, built on RDF
Free choice of (model-based) syntax