Towards an Interoperability Framework for Metadata Standards

Presenter: Mikael Nilsson <mini@nada.kth.se>

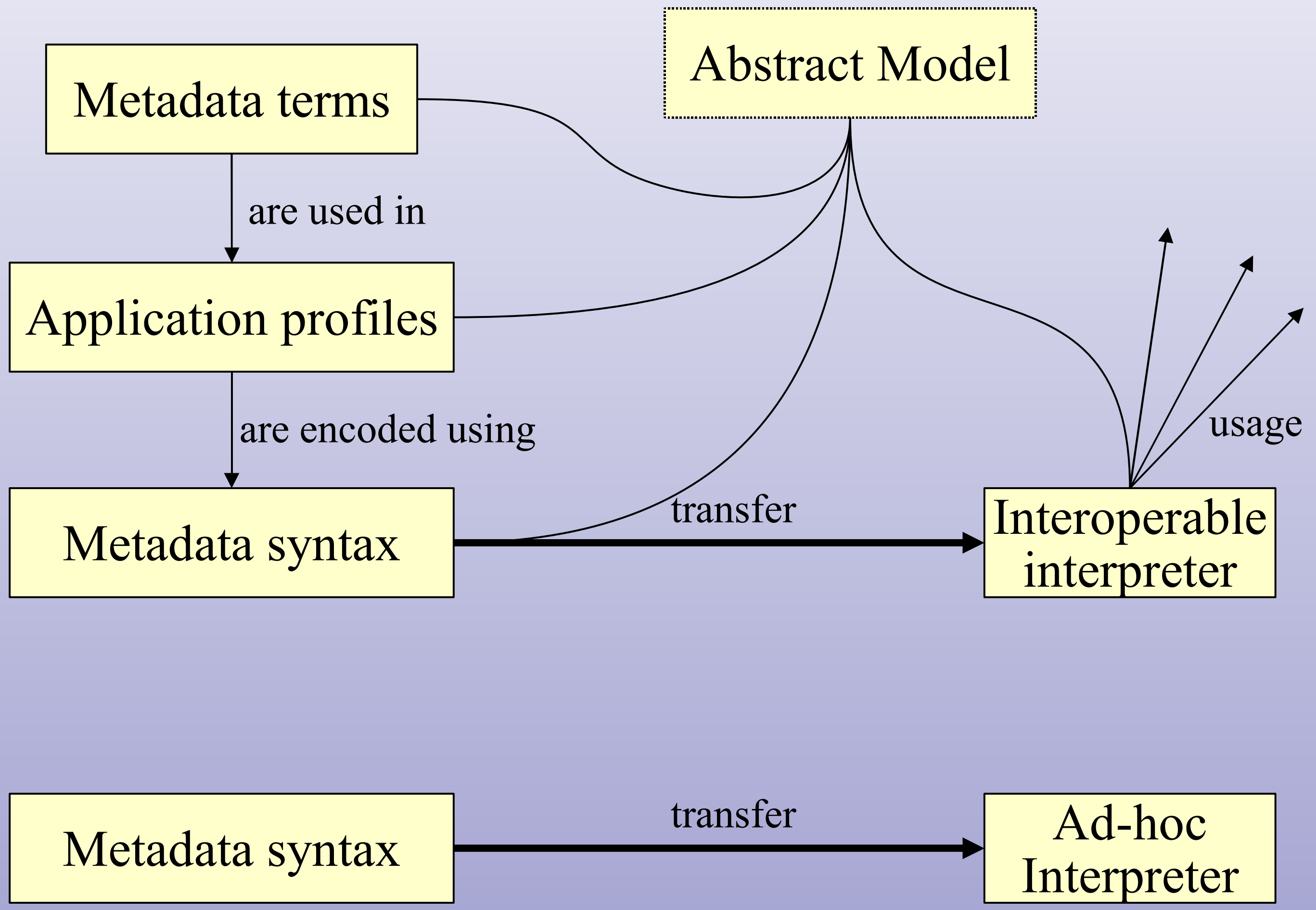
Co-authors:

Pete Johnston <pete.johnston@eduserv.org.uk>
Ambjörn Naeve <amb@nada.kth.se>
Andy Powell <andy.powell@eduserv.org.uk>

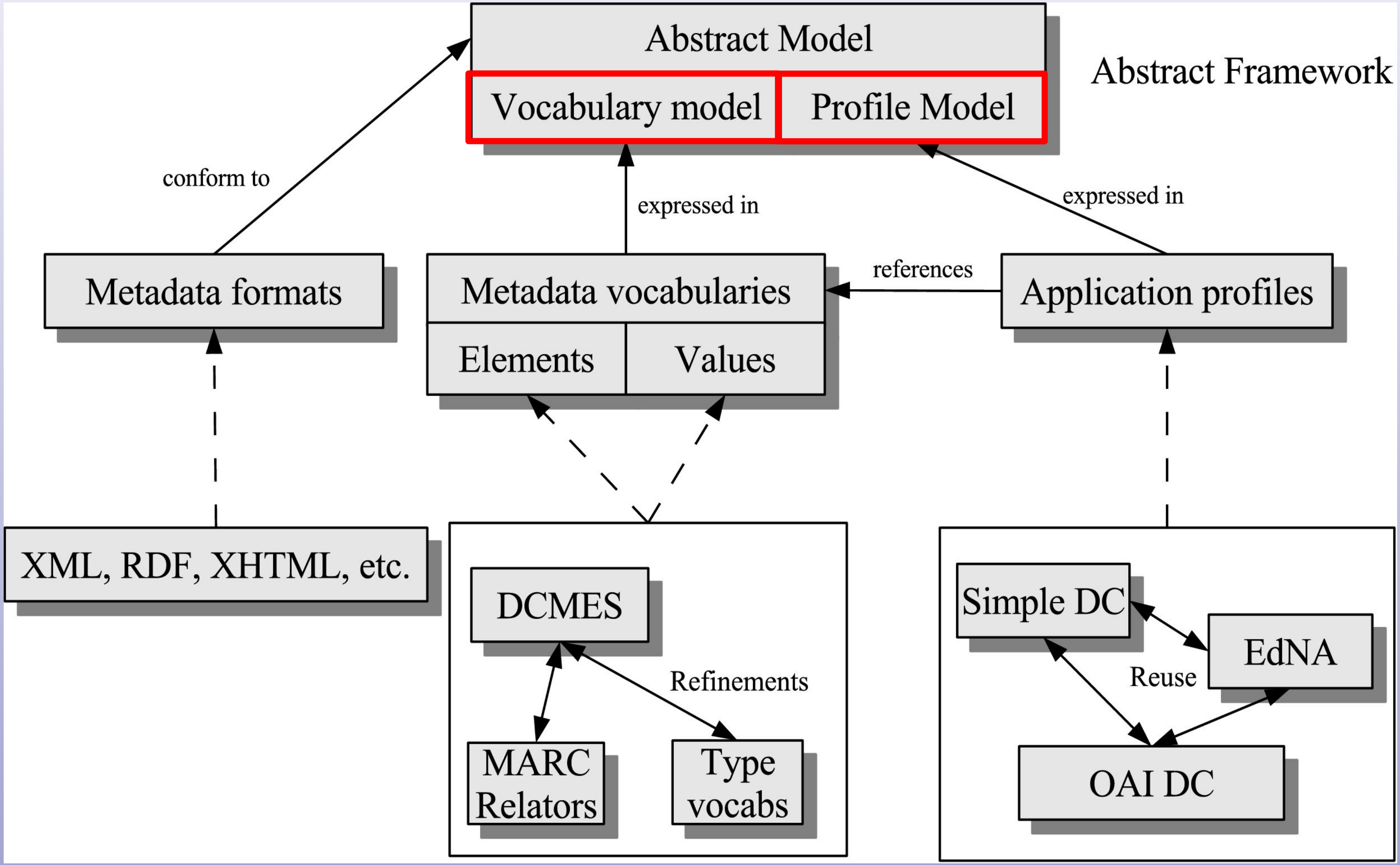
Interoperability for DC metadata

- Reusing existing properties...
  - in new application profiles
  - in new syntaxes
- Adding new properties to...
  - existing application profiles
  - existing syntaxes
- Introducing new values to existing properties
- Expressing existing APs in new syntaxes

Abstract information flow

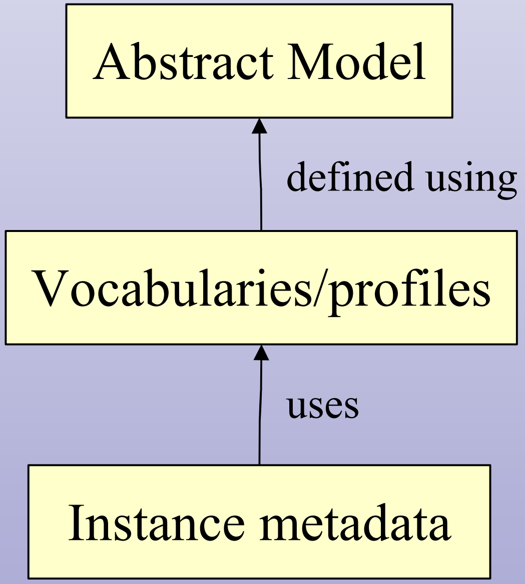


The Dublin Core Framework



DCMI Abstract Model

- Refined and formalized version of the “DCMI Grammatical Principles”



- Defines the core constructs used in DC metadata: *properties*, *vocabulary encoding schemes*, etc…
- Basis for defining vocabularies, profiles and syntaxes

Vocabulary Model (elements)

Example: http://dublincore.org/documents/dcmi-terms/

Each term is specified with the following minimal set of attributes:

Name:	The unique token assigned to the term.
URI:	The Uniform Resource Identifier used to uniquely identify a term.
Label:	The human-readable label assigned to the term.
Definition:	A statement that represents the concept and essential nature of the term.
Type of Term:	The type of term, such as Element or Encoding Scheme, as described in the DCMI Grammatical Principles.
Status:	Status assigned to term by the DCMI Usage Board, as described in the DCMI Usage Board Process.
Date issued:	Date on which a term was first declared.

Where applicable, the following attributes provide additional information about a term:

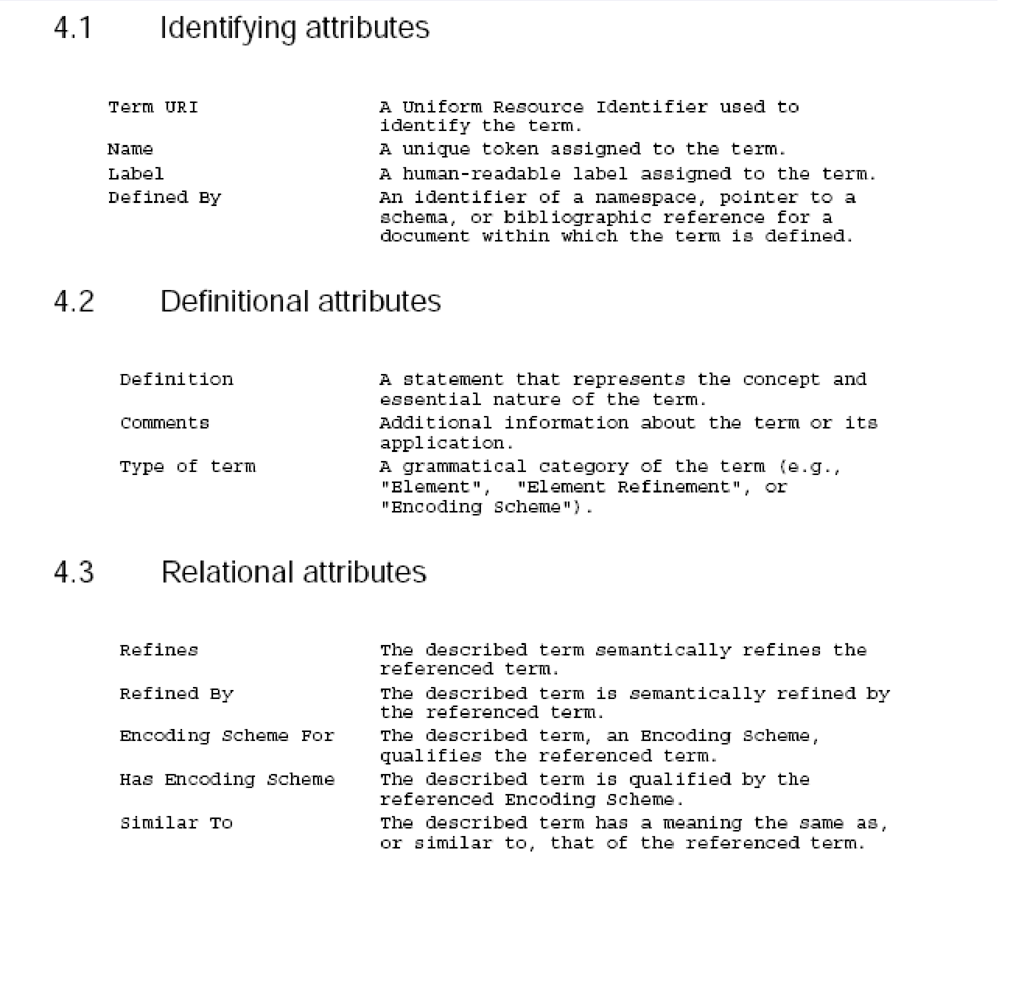
Comment:	Additional information about the term or its application.
See:	A link to authoritative documentation.
References:	A citation or URL of a resource referenced in the Definition or Comment.
Refines:	A reference to a term refined by an Element Refinement.
Qualifies:	A reference to a term qualified by an Encoding Scheme.
Broader Than:	A reference from a more general to a more specific Vocabulary Term.
Narrower Than:	A reference from a more specific to a more general Vocabulary Term.

Vocabulary Model (values)

- No real consensus on how to describe value vocabularies.
- Value vocabularies come in many kinds
  - simple lists
  - thesauri
  - taxonomies
  - ontologies
  - etc.
- SKOS?

Profile Model

Example: CEN/ISSS CWA 14855



4.1 Identifying attributes

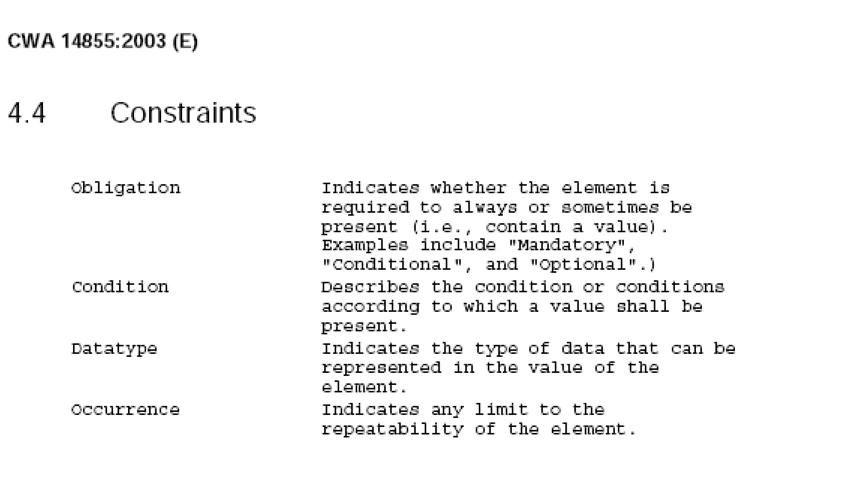
Term URI	A Uniform Resource Identifier used to identify the term.
Name	A unique token assigned to the term.
Label	A human-readable label assigned to the term.
Defined By	An identifier of a namespace, pointer to a schema, or bibliographic reference for a document within which the term is defined.

4.2 Definitional attributes

Definition	A statement that represents the concept and essential nature of the term.
Comments	Additional information about the term or its application.
Type of term	A grammatical category of the term (e.g., "Element", "Element Refinement", or "Encoding Scheme").

4.3 Relational attributes

Refines	The described term semantically refines the referenced term.
Refined By	The described term is semantically refined by the referenced term.
Encoding Scheme For	The described term, an Encoding Scheme, qualifies the referenced term.
Has Encoding Scheme	The described term is qualified by the referenced Encoding Scheme.
Similar To	The described term has a meaning the same as, or similar to, that of the referenced term.



CWA 14855:2003 (E)

4.4 Constraints

Obligation	Indicates whether the element is required to always or sometimes be present (i.e., contain a value). Examples include "Mandatory", "Conditional", and "Optional".)
Condition	Describes the condition or conditions according to which a value shall be present.
Datatype	Indicates the type of data that can be represented in the value of the element.
Occurrence	Indicates any limit to the repeatability of the element.

Where we are (DC family)

Framework concept	*Dublin Core framework*
Abstract Model	DCMI Abstract Model
Metadata Formats	XML, RDF and HTML bindings
Metadata Element Vocabularies	DCMES, large set of external properties and encoding schemes
Metadata Value Vocabularies	DCMIType vocabulary. Many external value vocabularies
Vocabulary Model	Not formalized, but some embryos for elements. SKOS for values?
Application Profiles	Some published by DCMI, many external application profiles
Profile Model	Not formalized, CWA 14885.

Where others are (IEEE LOM family)

Framework concept	*LOM framework*
Abstract Model	Implicit in LOM Data Model
Metadata Formats	XML binding
Metadata Element Vocabularies	LOM Data Model includes element vocabulary, various extensions to LOM
Metadata Value Vocabularies	LOM Data Model includes several basic value vocabularies, many external vocabularies
Vocabulary Model	Not formalized, IMS VDEX for values, element model difficult
Application Profiles	LOM Data Model includes basic application profile, many external application profiles.
Profile Model	Not formalized

Where others are (RDF family)

Framework concept	*Semantic Web framework*
Abstract Model	RDF Concepts and Abstract Syntax
Metadata Formats	RDF/XML syntax, N-triples, etc.
Metadata Element Vocabularies	Many external element vocabularies
Metadata Value Vocabularies	Many external value vocabularies
Vocabulary Model	RDF Vocabulary Description Language (elements and values)
Application Profiles	Many in the form of ontologies
Profile Model	Possibly OWL, the Web Ontology Language

Cross-framework interoperability

- How about interoperability between
  - DC <==> IEEE LOM <==> RDF <==> MPEG-7 etc.?
- All follow the same basic pattern
- BUT – very different abstract models
- In general: cross-framework is hard
- DC <==> RDF works – as models are intentionally compatible
- See also "*The Future of Learning Object Metadata Interoperability*", in Koohang A. (ed.) Learning Objects: Standards, Metadata, Repositories, and LCMS, in press.

Terminology

- Stop using “metadata standard” or “schema”
- Start using either
  - abstract model
  - metadata format/syntax
  - metadata vocabulary
  - application profile
- “What kind of specification am I trying to produce?”
  - Most: application profile (+ some vocabulary)

Take home message

- Formalized models (DCAM, vocabs, profiles) pave the way for ***interoperable processing***
- A clear framework helps us fill in the blanks
- Much progress over the last year! See DC-ARCH...
- The word ”schemas” is overused...