



# Application Profiles: A Tutorial

Diane I. Hillmann  
Cornell University



# Definitions & Goals

- What is an Application Profile?
- Why Application Profiles?
- Major issues in AP Development
- Communities & Process



# Application Profile: Definition

“A Dublin Core Application Profile (DCAP) is a declaration specifying which metadata terms an organization, information provider, or user community uses in its metadata. By definition, a DCAP identifies the source of metadata terms used—whether they have been defined in formally maintained standards such as Dublin Core, in less formally defined element sets and vocabularies, or by the creator of the DCAP itself for local use in an application. Optionally, a DCAP may provide additional documentation on how the terms are constrained, encoded or interpreted for application-specific purposes.” --*CEN CWA 14855:2003*



# Purpose of Application Profiles

- Document semantics and constraints for a particular set of instance metadata
- Provide focus and documentation of community consensus and intent
- Identify emerging semantics as possible candidates for formal standardization
- Provide guidance for semantic crosswalks and specifications for DTDs
- Serve as documentation for rules and criteria by which a specific set of metadata was created

-- *CEN CWA 14855:2003*



# Are Application Profiles only for DC?

- No, but DC has pushed the usage of APs and provided important guidelines
- CanCore is a good example of a non-DC AP (uses IEEE LOM)
- Other metadata schemas will provide different challenges



# Who is an AP for?

- Humans
  - “Principle of readability”: include enough information to be of optimal usefulness for its intended audience
  - Provide better quality control for metadata within and beyond specific domains or projects
- Machines
  - Machine-readable APs will make interoperability more achievable at lower cost
  - Increased precision when identifying terms with Uniform Resource Identifiers (URIs) brings us closer to the goals of the Semantic Web



# Why not use just one metadata standard?

- Different starting points
- Different functional requirements
- Different levels of granularity for different things
- Different “views” of reality
- **The days of “one size fits all” standards are over**
  - But domains are now overlapping and becoming “liquid”
- The challenge now is *interoperability* and *re-purposing*

*[Slide derived from presentation by Godfrey Rust, 5/2005]*



# Major AP Issues

- Human vs. machine needs
- Mixing and matching in a diverse metadata world
- Requirements for legitimate re-use
- Breaking new ground with new properties



# Components of an AP

- Human-readable documentation
  - Property descriptions and relationships
  - Domain specific instruction
  - Obligation and constraints
- Machine-readable versions being developed
  - Based on XML and RDF
  - Will enable improved functionality for quality control and analysis



# Using Properties from Other Schemas

- Determining reusability of terms
  - Is the term a real “property” and defined as such within the source schema?
  - Is the term declared properly, with a URI and adequate documentation and support?
  - In general, properties whose meaning is partly or wholly determined by its place in a hierarchy are not appropriate for reuse without reference to the hierarchy



# Using the “Term Decision Tree”

- Developed as a method for determining compliance with the DC Abstract Model
- Assists in identifying the appropriate “level” for a term: element, element refinement, encoding scheme
- Available at:  
<http://dublincore.org/architecturewiki/TermDecisionTree>



# Identification of Terms

- CORES Resolution (12/02): the preferred method of identification is a citation to a URI
- URIs are established for all DCMI terms, and are in the process for IEEE/LOM and some other metadata formats
- URIs will be required for terms in Application Profiles reviewed by the DC Usage Board



# Legitimizing re-use

- Take care, consider property definitions carefully
- Pay attention to the DCMI Abstract Model, and the developing DCMI Description Set Profile
- Document your choices and decisions for those who will need to clean up behind you!



# Creating new properties in an AP context

- Caveats
  - No firm rules yet!
- For the adventurous, here's a start at a path:
  - Defining new properties
  - Documenting new properties
  - Registration/Exposure




# Defining new properties

- Where to start:
  - Do your research! See what others have done, whether there are terms already in use
  - Enlist your community for guidance and support
  - Define Name, Label, Definition, Relationships, etc.
  - Determine a URI (implies a stable “home” for your terms)



# Documenting new properties

- Minimum: a web page or wiki, with the relevant information available to other implementations
- Better: a web page or wiki and an accessible XML schema or RDFS expression using your terms as part of your application profile



# Registering local value vocabularies

- Controlled vocabulary registries still in active development
  - As an example, see <http://metadataregistry.org>
- In the meantime, if using web pages:
  - Use standard thesaural structures and practices, such as those in NISO Z39.19 – 2003 (<http://www.niso.org/standards/resources/Z39-19.pdf>) or SKOS
  - Plan for the sustainability of the vocabulary over time!



# Getting Started With Your AP

- Determining AP scope and purpose
- Choosing a basic metadata format
- Attributes for describing terms
- Setting up documentation, decision making and community review processes
- Maintaining realistic expectations
- Standardization/review



# Scope and Purpose

- What is the context of the Application Profile?
- What are the community or project boundaries and goals?
- How is the community organized? Does it already exchange metadata?
- What are the community's unmet needs?
- Is there a pre-existing communication forum? A group of metadata-aware practitioners?



# Making Metadata Format Choices

- Look at what others in the domain are using (Does not have to be DC)
- Consider:
  - stability/volatility of the standard (and whether it really IS a standard)
  - how the community for the standard integrates new needs and ideas
  - startup and maintenance costs for use in an individual project (higher for more complex formats and implementations)
- Document choices and reasoning for your successors (they will thank you)



# Organizing Communities

"Taking OAI and DC 'off the shelf' as proven standards having widespread acceptance in the digital libraries community was decisive, permitting OLAC to unite disparate subcommunities and reach consensus. In particular, DC was simple, applicable to all kinds of resources, and widely used outside our community. Had we come to our first workshop with the proposal that the community needed to invent a metadata standard, all our resolve would have dissipated in factionalism. Thus, not only was DC both simple and mature, it was also a political expedient."

--Steven Bird and Gary Simons, *Building an Open Language Archives Community on the DC Foundation* (in "Metadata in Practice," ALA Editions, 2004)



# Looking at your Community

- Who are the target users?
- Who are the stakeholders in the community
- What resources are available for the task?
  - Communications methods
  - People
- What are the political realities?




# Maintaining Realistic Expectations

- Creating an Application Profile takes time, requires organizational effort and persistence
- There are still open questions on implementation and syntax—which you should be prepared to learn about
- Begin by focusing on community organization and examining already available examples to guide your consensus building activities
- Be aware of developments in technical areas as you go--they will have an impact on your work!



# Standardization/Review

- For APs based on Dublin Core, the DC Usage Board is providing review--processes for doing so in active development
- DC UB reviews will provide guidance on DC compliance primarily
- Work is ongoing on including APs as part of distributed registries, and in creating wiki templates as a tool for creating machine-readable expressions



# Example: Collection Description AP

- Developed by the DC Collection Description Working Group
- Recently reviewed by the DC Usage Board
  - Determined to be “conformant” with the DC Abstract Model
  - Available at:  
<http://dublincore.org/groups/collections/collection-application-profile/>

# Namespaces used in Collection Description AP

Vocabulary Title	Namespace Name	Prefix
The Dublin Core Metadata Element Set, v1.1	<a href="http://purl.org/dc/elements/1.1/">http://purl.org/dc/elements/1.1/</a>	dc
Dublin Core Terms	<a href="http://purl.org/dc/terms/">http://purl.org/dc/terms/</a>	dcterms
Dublin Core Type Vocabulary	<a href="http://purl.org/dc/dcmitype/">http://purl.org/dc/dcmitype/</a>	dcmitype
MARC Relator Code Properties	<a href="http://www.loc.gov/loc/terms/relators/">http://www.loc.gov/loc/terms/relators/</a>	marcrel
Collection Description Terms (collection-specific terms)	<a href="http://example.org/cld/terms#">http://example.org/cld/terms#</a> [temporary URI, final URI to be confirmed]	cld
General Description Terms (non-collection-specific terms)	<a href="http://example.org/gen/terms#">http://example.org/gen/terms#</a> [temporary URI, final URI to be confirmed]	gen
Collection Description Type Vocabulary Terms	<a href="http://example.org/cld/cdtype#">http://example.org/cld/cdtype#</a> [temporary URI, final URI to be confirmed]	cldtype

<b>Term Identifier</b>	<a href="http://purl.org/dc/terms/provenance">http://purl.org/dc/terms/provenance</a>								
Qualified Name	dcterms:provenance								
Defined By	Dublin Core Terms <a href="http://purl.org/dc/terms/">http://purl.org/dc/terms/</a>								
Type of Term	Property (Element)								
Source Label	Provenance								
Label in this DCAP	Custodial History								
Source Definition	A statement of any changes in ownership and custody of the resource since its creation that are significant for its authenticity, integrity and interpretation.								
Usage in this DCAP	A statement of any changes in ownership and custody of the collection that are significant for its authenticity, integrity and interpretation.								
Comments for this DCAP	[n/a]								
Refines	[n/a]								
Refined by	[n/a]								
Uses Vocabulary Encoding Scheme	[Value type not specified]								
	<table border="1"> <thead> <tr> <th>Value URI</th> <th>Value String</th> <th>Syntax Encoding Scheme(s)</th> <th>Rich Represent</th> </tr> </thead> <tbody> <tr> <td>Optional</td> <td>Mandatory</td> <td></td> <td>Not permitted</td> </tr> </tbody> </table>	Value URI	Value String	Syntax Encoding Scheme(s)	Rich Represent	Optional	Mandatory		Not permitted
Value URI	Value String	Syntax Encoding Scheme(s)	Rich Represent						
Optional	Mandatory		Not permitted						
Obligation	Optional								
Condition	[n/a]								
Occurrence	Minimum: 0, Maximum: unbounded								

# Identifying Attributes

Term Identifier	<a href="http://purl.org/dc/terms/provenance">http://purl.org/dc/terms/provenance</a>
Qualified Name	dcterms:provenance
Defined By	Dublin Core Terms <a href="http://purl.org/dc/terms/">http://purl.org/dc/terms/</a>
Source Label	Provenance
Label in this DCAP	Custodial History

- First four attributes come directly from DCMI
- Greyed attribute “Label in this DCAP” specifies the label that this community prefers for this attribute

# Definitional Attributes

Type of Term	Property (Element)
Source Definition	A statement of any changes in ownership and custody of the resource since its creation that are significant for its authenticity, integrity and interpretation.
Usage in this DCAP	A statement of any changes in ownership and custody of the collection that are significant for its authenticity, integrity and interpretation.
Comments for this DCAP	[n/a]

- “Type of term” and “Source Definition” are from DCMI term declaration
- “Usage in this DCAP” and “Comments for this DCAP” are added (or not) by the community defining the Application Profile

# Relational Attributes

Refines	[n/a]			
Refined by	[n/a]			
Uses Vocabulary Encoding Scheme	[Value type not specified]			
	<b>Value URI</b>	<b>Value String</b>	<b>Syntax Encoding Scheme(s)</b>	<b>Rich Represent</b>
	Optional	Mandatory		Not permitted

- Because “Type of Term” for this property is *element* it cannot refine another term; this term has no refinements
- No Vocabulary encoding scheme is recommended; a value string representation is mandated

# Constraints & Obligations

Obligation	Optional
Condition	[n/a]
Occurrence	Minimum: 0, Maximum: unbounded

- Because use of the element is Optional in this AP, no occurrence is required, but any number of iterations is allowed
- No special conditions are imposed

<b>Term Identifier</b>	<a href="http://www.loc.gov/loc.terms/relators/OWN">http://www.loc.gov/loc.terms/relators/OWN</a>		
Qualified Name	marcrel:OWN		
Defined By	MARC Relator Codes <a href="http://www.loc.gov/loc.terms/relators/">http://www.loc.gov/loc.terms/relators/</a>		
Type of Term	Property (Element)		
Source Label	Owner		
Label in this DCAP	Owner		
Source Definition	The person or organization that currently owns an item or collection.		
Usage in this DCAP	An entity who has legal possession of the collection.		
Comments for this DCAP	[n/a]		
Refines	[n/a]		
Refined by	[n/a]		
Uses Vocabulary Encoding Scheme	[Value type not specified]		
	<b>Value URI</b>	<b>Value String</b>	<b>Syntax Encoding Scheme(s)</b>
	Optional	Mandatory	Not permitted
Obligation	Optional		
Condition	[n/a]		
Occurrence	Minimum: 0, Maximum: unbounded		

# ID & Definition: Non-DCMI Term

Term Identifier	<a href="http://www.loc.gov/loc.terms/relators/OWN">http://www.loc.gov/loc.terms/relators/OWN</a>
Qualified Name	marcrel:OWN
Defined By	MARC Relator Codes <a href="http://www.loc.gov/loc.terms/relators/">http://www.loc.gov/loc.terms/relators/</a>
Type of Term	Property (Element)
Source Label	Owner
Label in this DCAP	Owner
Source Definition	The person or organization that currently owns an item or collection.
Usage in this DCAP	An entity who has legal possession of the collection.

- Term originates in LC namespace but is NOT able to be dumbed down to Contributor
- Still includes information from the originating term declaration PLUS specific community usage

## Item Type [cld:itemType]

<b>Term Identifier</b>	<a href="http://example.org/cld/terms#itemType">http://example.org/cld/terms#itemType</a>										
Qualified Name	cld:itemType										
Defined By	Collection Description Terms <a href="http://example.org/cld/terms#">http://example.org/cld/terms#</a>										
Type of Term	Property (Element)										
Source Label	Item Type										
Label in this DCAP	Item Type										
Source Definition	The nature or genre of the content of one or more items within the collection.										
Usage in this DCAP	[n/a]										
Comments for this DCAP	[n/a]										
Refines	[n/a]										
Refined by	[n/a]										
Uses Vocabulary Encoding Scheme	dcterms:DCMIType, Dublin Core Terms <a href="http://purl.org/dc/terms/DCMIType">http://purl.org/dc/terms/DCMIType</a> <table border="1" data-bbox="462 915 1454 1086"> <thead> <tr> <th>Value URI</th> <th>Value String</th> <th>Syntax Encoding Scheme(s)</th> <th>Rich Represent</th> </tr> </thead> <tbody> <tr> <td>Optional</td> <td>Mandatory</td> <td></td> <td>Not permitted</td> </tr> </tbody> </table>			Value URI	Value String	Syntax Encoding Scheme(s)	Rich Represent	Optional	Mandatory		Not permitted
Value URI	Value String	Syntax Encoding Scheme(s)	Rich Represent								
Optional	Mandatory		Not permitted								
Obligation	Optional										
Condition	[n/a]										
Occurrence	Minimum: 0, Maximum: unbounded										

# New Term Defined

<b>Property URI</b>	<b><a href="http://purl.org/cld/terms/itemType">http://purl.org/cld/terms/itemType</a></b>
Qualified Name for Property	cld:itemType
Defined By	Collection Description Terms <a href="http://purl.org/cld/terms/">http://purl.org/cld/terms/</a>
Type of Term	Property
Subproperty Of	[n/a]
Source Label	Item Type
Label in this DCAP	[n/a]
Source Definition	The nature or genre of one or more items within the collection.

- Property URI declared using PURL
- Since this term is new, all attributes are specific to the community, none are *greyed*

# New Term Definition, cont.

Usage in this DCAP	[n/a]											
Comments for this DCAP	<p>A value string must be provided; a value URI may also be provided.</p> <p>Where the collection includes items of multiple types, a separate statement should be used for each type which is regarded as significant for a user of the collection description.</p>											
Uses Vocabulary Encoding Scheme	<p><a href="#">dcterms:DCMIType</a>, Dublin Core Terms  <a href="http://purl.org/dc/terms/DCMIType">http://purl.org/dc/terms/DCMIType</a></p> <table border="1"> <thead> <tr> <th>Value URI</th> <th>Value String</th> <th>Syntax Encoding Scheme(s)</th> <th>Rich Represent</th> </tr> </thead> <tbody> <tr> <td>Optional</td> <td>Mandatory</td> <td></td> <td>Not permitted</td> </tr> </tbody> </table>				Value URI	Value String	Syntax Encoding Scheme(s)	Rich Represent	Optional	Mandatory		Not permitted
Value URI	Value String	Syntax Encoding Scheme(s)	Rich Represent									
Optional	Mandatory		Not permitted									
Obligation	Optional											
Condition	[n/a]											
Minimum Occurrences	0											
Maximum Occurrences	unbounded											

- Note the use of DCMIType vocabulary for this new term
- DCMIType vocabulary is also specified for Type in this DCAP



# Where Are APs Going?

- Currently an emphasis on machine-readable Aps
  - The new Description Set Profile is an important step in accomplishing this goal
  - From the Dublin Core Abstract Model:  
*"A description set is a set of one or more descriptions, each of which describes a single resource."*

<http://dublincore.org/architecture/wiki/DescriptionSetProfile>



# What Can a Description Set Profile Do?

- A DSP is a way of describing structural constraints on a description set. It constrains the resources that may be described by descriptions in the description set, the properties that may be used, and the ways a value surrogate may be given.
- A DSP can be used for many different purposes, such as:
  - as a formal representation of the constraints of a Dublin Core Application Profile
  - as configuration for databases
  - as configuration for metadata editing tools



# A DSP Does Not:

- Provide human-readable documentation
- Provide definition of vocabularies
- Provide version control
  - A DSP contains the formal syntactic constraints only, and will need to be combined with human-readable information, usage guidelines, version management, etc. in order to be used as an application profile.



# Thank you!

## Questions?

Please feel free to provide feedback to:

Diane Hillmann  
dih1@cornell.edu