DCMI-Tools: Ontologies for Digital Application Description

Jane Greenberg[1]*; Thomas Severiens*[2]

[1] School of Information and Library Science, University of North Carolina in Chapel Hill, USA
e-mail: janeg@ils.unc.edu
[2] Institut für wissenschaftliche Information e.V., Universität Osnabrück, Germany
e-mail: severiens@mathematik.uni-osnabrueck.de

Abstract

The growth in electronic and digital publishing on the World Wide Web has led to the development of a wide range of tools for generating metadata. As a result, it can be difficult to select the appropriate type of application and the best metadata tool to support a project's metadata needs. The Dublin Core Tools (DCMI Tools) Community recognizes this need and is developing an application profile and a taxonomy of tool functionalities for describing metadata applications. The community will use the application profile and the taxonomy to standardize access to information on metadata via the DCMI Tools and Software program. This paper reports on the DCMI Tool Community's activities to develop an application profile for describing the wide range of applications (algorithms; metadata templates, editors, and generators; and other software) fitting this rubric. The paper begins with an introduction to metadata application challenges, and introduces the DCMI Tools Community in order to provide important historical context. Next, the paper reviews the concept of application profile and emphasizes the importance of this approach for describing metadata tools. The paper reviews procedures to develop the application profile and presents the DCMI Tools application profile. The paper also presents a metadata tool functionality taxonomy (to be used with the application profile), a glossary (to assist people in learning about metadata tools), and the DCMI Tool Community's implementation plans. The final part of the paper presents several conclusions and highlights next steps.

Keywords: metadata tools; application profile; DCMI Tools Community

1 Introduction

Today's metadata tool environment includes offerings ranging from algorithms that plug in to various multi-functional software applications to fully developed tools specifically labeled as metadata editors, templates, and generators. Included in the mix are many software applications, such as word processing and publication software (e.g., Microsoft's WORD and Acrobats Adobe) and MP3 software that increasingly include functionalities supporting metadata generation. Tools in this category often include templates for storing summary metadata, such as "keywords" or "author name" or a brief "description". This type of software generally automatically generates a range of metadata, such as "date created", "date last modified", "size" and "format" [1]. There is also an evolution of blog software and social software (e.g., Flickr or Del.icio.us) supporting similar metadata generation, including tags. Metadata generated with any of these applications (designated metadata tools, software applications, and social software) can be harvested by metadata tools to create coherent or more substantial metadata records, which can be ported into a metadata repository to support resource discovery and other desired metadata functionalities [2].

Although these developments are exciting, they have complicated our view of the metadata tool landscape. That is the wide range and diversity of applications can make it difficult to select the appropriate type of application and the best metadata application to support a project's metadata needs. Should a digital library project invest in a fully functional off-the-shelf metadata generation application? What open source algorithms might be accessible that could be integrated with an institutions existing software suite to satisfy metadata needs? Catalogers, metadata professionals, information architects, and project managers are constantly asking these and other questions to determine which applications will suit their needs. Their inquiry is made difficult because of the absence of a single place providing unified and consistent descriptions of metadata tools.

The Dublin Core Tools (DCMI Tools) Community, a part of the Dublin Core Metadata Initiative (DCMI), is addressing this challenge [3, 4]. For the last several years this community has provided a Web page with access information and brief descriptions of applications supporting the generation of Dublin Core metadata records. As the metadata tool community has grown to include both developers and users, so too has the need to provide

unified and collective information about metadata applications. The need expands beyond applications supporting Dublin Core metadata, to tools supporting metadata creation following:

- Standard schemes beyond the Dublin Core (e.g., ONIX or the EAD).
- Content value standards (e.g., *Library of Congress Classification* system) and authority files.
- Encoding schemes to standardize the use of content value standards even further (e.g., W3C Date Time Format standard).

The DCMI Tools Community is addressing this need via the development of an application profile and a taxonomy of tool functionalities—both of which can be used for describing metadata applications generally accessible for digital library and related initiatives.

This paper reports on the DCMI Tools activities to develop an application profile for describing the wide range of applications (algorithms; metadata templates, editors, and generators; and other software) fitting this rubric. The following sections of this paper are ordered as follows: section 2 introduces the DCMI Tools Community and provides some historical context; section 3 reviews the concept "application profile" and emphasizes why this approach supports a unified description of metadata tools; section 4 presents procedures to develop the current DCMI Tools application profile; section 4 presents the DCMI Tools application profile, section 5 presents a taxonomy of tool functionalities for classifying metadata applications, a glossary containing terminology that is important for the metadata tool community, and application profile implementation steps; section 6 includes several conclusions and next steps.

2 The DCMI Tools Community

The DCMI Tools Community is a "forum for individuals and organizations involved in the development and usage of tools and applications based on Dublin Core Metadata or other metadata standards that interoperate with and enhance functionality of the Dublin Core" [5]. The DCMI Tools community was initially a working group and was initiated at the 1999 Dublin Core conference in Frankfurt, Germany. The founding chairs were Roland Schwänzl (Osnabrück University) and Harry Wagner (OCLC). The working group initially focused on RDF-Tools and XML-Schema, as well as on DAML+OIL (which since that time has developed as SOAP web-services). At the outset, the DCMI Tools WG recognized the metadata community's need to access information about metadata applications. The Tools WG, therefore, took up the initiative of documenting and making accessible basic and important information about metadata applications via the DCMI Website through the "Tools and Software" program [6].

Although no formal descriptive standard was created to describe the applications, a broad taxonomy was developed to classify the range of applications being represented. Metadata tools being currently represented via DCMI's Tools and Software program are classed accordingly: Utilities, Creating Metadata (Templates), Tools for the Creation/Change of Templates, Automatic Extraction/Gathering of Metadata, Automatic Production of Metadata, Conversion Between Metadata Formats, Integrated (Tool) Environments, Application Profiles (Examples and Tools), and Metadata Search Engines. Details given for the tools represented via this site range from brief abstracts to more descriptive accounts documenting the metadata elements and schemes a tool supports.

During the Dublin Core 2006 conference in Manzanillo, Colima, Mexico, the DCMI Tools working group was transformed to what the DCMI refers to as a community [5]. The goal of a DCMI community is to facilitate the "exchange of information, general discussion within a specific area of interest" [3]. This change was very timely for the DCMI Tools WG, which had a year earlier Madrid, Spain, revised their charge to develop as a forum for two classes of users: tool developers and individuals interested in using tools. The DCMI Tools working group sponsored a workshop at the 2006 Joint Conference on Digital Libraries (JCDL), bringing together these users into a single community [7]. These developments, and the growing interest in metadata tools well beyond the immediate DCMI community, have motivated the reevaluation of the current classification of tools represented via the DCMI Tools and Software program [6]. This work has been a major focus of the DCMI Tools community via the last year, through a task group comprised of the DCMI Tools community co-chairs, with input from other members of the DCMI tools community. Our process of revision has required the creation of an application profile. The next section of this paper defines what an application profiles is, and why we selected this approach.

3 Application Profiles: A Practical Approach for Describing Metadata Tools

An application profile is a declaration of the metadata terms an organization, information resource, application, or user community uses in its metadata. In a broader sense, an application profile includes the set of metadata elements, policies, and guidelines defined for a particular application or implementation. The elements may be from one or more element sets, thus allowing a given application to meet its functional requirements by using metadata elements from several element sets including locally defined sets. For example, a given application might choose a specific subset of the Dublin Core elements that meets its needs, or may include elements from the Dublin Core, another element set, and several locally defined elements, all combined in a single schema, as for example the "Dublin Core Collection Description Application Profile" [8] does. An application profile is not considered complete without documentation that defines the policies and best practices appropriate to the application.

Application profiles are created for practical reasons. *First*, it makes no sense to reinvent the wheel. Why should a project create a metadata scheme from scratch, when there is already a scheme, or a series of schemes, that have already defined needed metadata elements, including implementation and use requirements? *Second*, it is recognized that often, a single scheme may not fully satisfy the needs of an individual initiative. For example, the Dublin Core metadata scheme is very useful for supporting resource discovery of digital resources in a digital library, although the elements do not adequately document and help manage resource preservation. A digital library wanting to facilitate the functions of both "resource discovery" and "preservation" might create an application profile, integrating elements from both the Dublin Core metadata standard and the PREMIS metadata standard [9]. *Third*, an application profile, pulling together elements from other schemes, facilitates greater interoperability on the World Wide Web. The reasons stated here, the growth in the availability of metadata tools, and the expansion of the DCMI Tools community, have motivated the development of the DCMI Tools application profile for representing algorithms, code pieces, and software tools. The next section of this paper presents a more detailed description on the methodology used generating the application profile.

4 Procedures for Developing the DCMI Tools Application Profile

The objective of our application profile activity is to describe algorithms, crosswalks, software, software-tools, and utilities collected in www.dublincore.org/tools/ in a coherent way. In moving forward, we have aimed to achieve best practices, resulting from discussion with participants from various fields. Our procedure has included the following steps:

1. An assessment of all elements in the available from the Dublin Core (ISO 15836-2003), DCTERMS (www.dublincore.org/documents/dcmi-terms/), and DOAP (Description of A Project) (usefulinc.com/doap/) and their applicability to the DCMI Tools community's goals. We selected these three schemes to cover the obvious needs for describing those applications being collected in our initial repository;

2. An initial three level ranking of each element's usefulness to our goals, with level *one* being necessary, *two* being potentially valuable *three* being not germane;

3. The composition of a DCMI Tools application profile, which included all level one ranked elements, and slightly over half of the level two items;

4. The development of a taxonomy of metadata tool functionalities—to be used with the application profile and for classifying metadata tools;

5. The development of a glossary to aide with tool classification and to facilitate communication among the metadata tool user community.

5 DCMI Tools Application Profile

The DCMI Tools application profile contains 17 elements, drawing from the Dublin Core, the DCTerms, and DOAP schemes. Nine of these elements contain qualifiers. Qualifiers can refine the meaning of an element, indicate where the value associated with an element came from, or the content formatting of an element (e.g, the format of year-month-date: YYYY-MM-DD). Table 1 presents an overview of our application profile, including examples for two applications, DC-dot and Picard Tagger.

Name-space	Element	Qualifiers	Example DC-dot	Example Picard Tagger
dc	contributor	doap:maintainer doap:developer doap:documenter doap:translator doap:tester	Rachel Heery	developer: LukasLalinsky developer: RobertKaye
dc	creator		Andy Powell	
dc	date	dcterms:created dcterms:dateCopyrighted dcterms:modified dcterms:issued	Created: 7 July 1997	issued: 2006-06-25
dc	description		Extracts and validates metadata from HTML resources and MS Office files. The generated metadata can be edited using the form provided and converted to various other formats (USMARC, SOIF, IAFA/ROADS, TEI headers, GILS, IMS or RDF) if required.	PicardTagger allows you to automatically look up the releases/tracks in your music collection and then write clean metadata tags (ID3 tags, Vorbis comment fields, etc.) to your files. It also allow syou to specify how and where to write cleanly tagged files to your hard drive.
dc	identifer	doap:repository	http://www.ukoln.ac.uk/metadata/dcdot/	http://musicbrainz.org/doc/PicardTagger repository: http://svn.musicbrainz.org/picard
dc	language		en-us, en-GB	
dc	publisher			
dc	relation	dcterms:hasPart dcterms:hasVersion dcterms:isPartOf dcterms:isReplacedBy dcterms:isRequiredBy dcterms:isVersionOf dcterms:replaces dcterms:requires doap: release	requires: Libwww-perl, soif.pl, Jon Knight's MARC module	requires: PyQt4 Mutagen (1.7) python-musicbrainz2 isPartOf: https://musicbrainz.helixcommunity.org/ release: 0.7.1
dc	rights	dcterms:accessRights dcterms:license	accessRights: open source license: http://www.gnu.org/copyleft/gpl.html	accessRights: open source license: http://www.gnu.org/copyleft/gpl.html
dc	rightsHolder			
dc	source	dcterms:URI		Workman, http://musicbrainz.org/doc/Workman
dc	title	dcterms:alternative	DC-dot	Picard Tagger
dc	type	dcterms:dataset dcterms:InteractiveResource dcterms:service dcterms:software	dcterms:InteractiveResource	dcterms:software
dcterms	audience	dctools:developer dctools:users		dctools:users dctools:developer
doap	location		Bath, UK	
doap	programming-language		Perl	Python
doap	operating-system			

Table 1: DCMI Tools Application Profile

6 A Taxonomy of Metadata Tool Functionalities

The application profile can be implemented within a semantic web framework, with automatic processes and requires the use taxonomy terms wherever possible. This will improve the representation of objects described, allowing for fairly complete the metadata descriptions. The most important part of the application profile is the classification of objects by genre, represented in our taxonomy.

Every object described may be in one or more of the following classes, which allows for sorting of tools by functionalities:

- Conversion
- Crosswalk
- Metadata Creation
- Metadata Encoding
- Metadata Extraction
- Metadata Generation
- Metadata Harvesting
- Metadata Templates
- Search Engines
- Translation
- Transliteration
- Validation

We will extend these classes as new types of software are developed. Classes not filled with latest software will be deleted, and the list will be revised as needed to allow for appropriate growth. We see this lists as being organic—in order to meet the needs of the tools community over time.

Some still open questions remain as part of our work in developing the profile. For example, location information requires additional attention. The most useful and precise approach is to give geographical coordinates, so a service can link to map serves. An alternative approach is to use a controlled vocabulary for geographic names. In this case, it would be desirable to allow for access and linking via international names (e.g., “Wien” (German version) versus “Vienna” (English version) versus “Wenen” (Dutch version). For the agent roles in the application profile we tried to use the roles defined in DOAP namespace (usefulinc.com/doap/) mostly reused from the foam-project results:

- developer
- documenter
- maintainer
- tester
- translator

To re-use the collected information in multiple frameworks, it will be requested to clearly define all vocabulary used. For use in semantic web framework this will be offered as RDFS, for human readability we restrict to textual representation in this article.

To assist with our work and further bring the metadata tool user community together, we have also developed a Glossary. This is presented in Table 2. The glossary is a new development produced by the DCMI Tools Community, and will be enhanced and modified as we continue our work.

Algorithm
a finite set of well-defined instructions for accomplishing some task which, given an initial state, will terminate in a defined end-state. (Wikipedia)

Application Profile
an assemblage of metadata elements selected from one or more metadata schemas and combined in a compound schema. Application profiles provide the means to express principles of modularity and extensibility. The purpose of an application profile is to adapt or combine existing schemas into a package that is tailored to the functional requirements of a particular application, while retaining interoperability with the original base schemas. Part of such an adaptation may include the elaboration of local metadata elements that have importance in a given community or organization, but which are not expected to be important in a wider context. (Duval)

Conversion
can refer to either
- conversion between schemas
- conversion of encoding (x/html to xml)

Crosswalk
a semantic mapping of metadata elements across metadata schema specifications. Crosswalks permit searching across multiple databases that use different schemas (Greenberg)

Metadata
An item of metadata may describe an individual data item or a collection of data items. Metadata is used to facilitate the understanding, use and management of data. (Wikipedia)

Metadata Creation
creation of metadata can be either
- by professional metadata creators; these include catalogers, indexers, and database administrators
- by technical metadata creators; these include webmasters, data in-putters, paraprofessionals, encoders and other persons who create metadata and may have had basic training but not professional level training
- by content creators; people who create the intellectual content of an object and the metadata for that object
- by community / subject enthusiasts; people who have not had any formal metadata-creation training but have special subject knowledge and want to assist with documentation (Greenberg)

Metadata Encoding
the syntax or prescribed order for the elements contained in the metadata description (NISO)

Metadata Extraction
synonym to Metadata Harvesting

Metadata Generation
the act of creating or producing metadata. Metadata can be generated by people, tools and processes (Greenberg)

Metadata Harvesting
a technique for extracting metadata from individual repositories and collecting it in a central catalog (NISO)

Metadata Template
Metadata format designed for some specific use or subject. (Severiens)

Namespace
In XML, a namespace is a collection of names, identified by a URI reference, that are used in XML documents as element types and attribute names. In order for XML documents to be able to use elements and attributes that have the same name but come from different sources, there must be a way to differentiate between the markup elements that come from the different sources. (Webopedia.com)

Schema
In general terms, any organization, coding, outline or plan of concepts. In terms of metadata, a systematic, orderly combination of elements or terms. In terms of DCMI term declarations represented in XML or RDF schema language, schemas are machine-processable specifications which define the structure and syntax of metadata specifications in a formal schema language. In terms of an encoding scheme, is a set of rules for encoding information that supports a specific community of users. See also Encoding scheme. (DCMI)

Search Engine
A utility capable of returning references to relevant information resources in response to a query. (DCMI)

Software
consisting of programs, enables a computer to perform specific tasks (Wikipedia)

Software-Tool
small piece of software, designed for developmental and laboratorial use (Severiens)

Translation
the interpretation of the meaning of a text in one language and the production, in another language, of an equivalent text that communicates the same message. Translation between may also convert meaning between semantics or schemes. (Wikipedia, Severiens)

Transliteration
Conversion of names or text not written in the roman alphabet to roman-alphabet form. (AACR Glossary)

Utility
software program that functions for a particular purpose. (Wikipedia)

Validation
- validating that syntax of element contents is correct (e.g. YYYY-MM-DD)
- validating the encoding (e.g., XML)

Table 2: DCMI Tools Glossary

The database, from which www.dublincore.org/tools is being generated, contains the following structure:

- Title: corresponding to the dc.title field in the app. profile.
- URL: corresponding to the dc.identifier field in the app. profile.
- Description: corresponding to the dc.description field in the app. profile.
- Classification: used to sort the service into the different classes.
- Free/commertial: this field is corresponding to the dc.rights qualifier dcterms:accessRights
- Online/download/webservice: corresponding to the dc.type field in the app. Profile and its qualifiers dcterms.InteractiveResource / dcterms.software / dcterms.service, a tag for dcterms.dataset may be added, if an entry is being included into the database.
- Country: corresponding to the field doap.laocation.
- Comment: This field allows some free text comments.
- Provider: corresponding to the dc.publisher field in the app. profile

Based on application profile developments, our plan is to add the following fields to the database:

- Information on the contributors, which can be
  - developers
  - documenters
  - maintainers
  - testers
  - translators
- Information on the creator(s)
- Information on the dates associated with the object, like
  - the date of its creation,
  - date of its latest modification,
  - date it was issued,
  - or the date of its copyright notice
- Information on the language of the object
- Information on the relations of the object to other objects in the database
- Information on the license like
  - a link to the licence text,
  - information on the licence holder,
  - while the date of the licence was already given with the dates above.
- Information on the source, if they differ from the compiled resource
- Information on the used programming language, if a source is available
- Information on the operating systems requested for running the software, if its not an webservice or online service.

7 Conclusions and Next Steps

The experience of creating the DCMI Tools application profile has been fruitful and resulted in an application profile that is ready for implementation. The DCMI Tools Community will be meeting at the DCMI-2007 Conference in Singapore this August to update members on this work. Prior to this conference, we will be testing the application profile and revising the DC Tools and Software program [6]. Our implementation will allow us to evaluate the overall effectiveness of the DCMI Tools application profile and identify areas requiring attention and revision. We will use our time in Singapore to share our findings and discuss any other outstanding issues, such as integrating location vocabulary from doap:location field. We will then begin to work on a collection and maintenance policy plan for keeping the DC Tools program up-to-date.

Acknowledgements

We would like to acknowledge Tessa Sullivan, doctoral students, School of Information and Library Science, University of North Carolina at Chapel Hill, USA, for her work on creating a table for the DC Tools Application Profile.

Notes and References

[1] GREENBERG, J.; SPURGIN, K;.CRYSTAL, A. Functionalities for Automatic-Metadata Generation Applications: A Survey of Metadata Experts' Opinions. International Journal of Metadata, Semantics, and Ontologies, (2006) 1(1): 3-20.

[2] GREENBERG, J. Understanding Metadata and Metadata Schemes. Cataloging & Classification Quarterly, (2005) 40(3/4): 17-36.

[3] DCMI Tools Community: http://dublincore.org/groups/tools/.

[4] Dublin Core Metadata Initiative: http://dublincore.org/.

[5] DEKKERS, M. Operational aspects of DCMI Work structure: Communities and Task. Date Issued: 2006-12-18: Groups: http://dublincore.org/documents/workstructure/.

[6] Tools and Software: http://dublincore.org/tools/.

[7] GREENBERG, J.; SEVERIENS, T., Metadata Tools for Digital Resource Repositories, D-Lib Magazine, (2006), Volume 12, Number 7/8, DOI:10.1045/july2006-greenberg.

[8] Dublin Core Collection Description Application Profile: http://www.ukoln.ac.uk/metadata/dcmi/collection-application-profile/.

[9] PREMIS: Data Dictionary for Preservation Metadata: http://www.oclc.org/research/projects/pmwg/premis-final.pdf.