Abstract:
The DCMI (Dublin Core Metadata Initiative) Tools Community supports the development and maintenance of DCMI standards by developing, collecting and providing operational and prototype implementations of standards. The objective of the community is to facilitate the usage and development of applications and functions based on open standards. The DCMI Tools Community provides a forum for two classes of users: tool developers and individuals interested in using tools. The focus of the workshop was on the development of an ontology for describing algorithms, tools, and applications of diverse evolution state of development and developed in the context of semantic web and DCMI metadata definition.

A draft version of the Metadata application profile was discussed during the meeting. A presentation covered DCMI application profiles in general, the development of tools-description, and a glossary of defined terms. The following discussion showed the relevance and diversity of the problem and resulted in an advanced version of the application profile.

This report gives a compilation of the presented slides, the handout material and the results of the following discussion.
DCMI-Tools:

Ontologies for Digital Application Description

Jane Greenberg, SILS, UNC, USA
Thomas Severiens, IW1 Osnabrück e.V., Germany

What is DCMI?

„The Dublin Core Metadata Initiative (DCMI) is an organization dedicated to promoting the widespread adoption of interoperable metadata standards and developing specialized metadata vocabularies for describing resources that enable more intelligent information discovery systems“

Taken from www.dublincore.org
DCMI Mission and Scope

The Dublin Core Metadata Initiative provides simple standards to facilitate the finding, sharing and management of information.

DCMI does this by:

• Developing and maintaining international standards for describing resources
• Supporting a worldwide community of users and developers
• Promoting widespread use of Dublin Core solutions

DCMI Characteristics (the 3 „I“s)

• **Independent:** DCMI is not controlled by specific commercial or other interests and is not biased towards specific domains nor does it mandate specific technical solutions

• **International:** DCMI encourages participation from organizations anywhere in the world, respecting linguistic and cultural differences

• **Influenceable:** DCMI is an open organization aiming at building consensus among the participating organizations; there are no prerequisites for participation
DCMI Communities & Task Groups

- DCMI Architecture Forum
- DCMI Accessibility Task Group
- DCMI Agents Task Group
- DCMI Collection Description Application Profile Task Group
- DCMI Education Application Profile Task Group
- DCMI Government Application Profile Task Group
- DCMI Kernel Task Group
- DCMI Libraries Application Profile Task Group
- DCMI RDF Task Force
- Joint DCMI/IEEE LTSC Taskforce
- Date Working Group

ELPUB 2007, Wien (Vienna)
DCMI-Tools Workshop

June 13, 2007
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.

• The objective of the Community is to facilitate the usage and development of applications and functions based on open standards. The DCMI Tools Community provides a forum for two classes of users: tool developers and individuals interested in using tools.

DCMI-Tool: Some History...

• Founded June 20, 2001 in Osnabrück (by Roland Schwänzl and Harry Wagner)

• Workshop March 2003 in Osnabrück

• Refounded in March 2005 in Göttingen (by Harry Wagner and Thomas Severiens)

• Workshops:
  – 03/2005: Göttingen, Germany
  – 09/2005: Madrid, Spain (DC 2005) (Chairs: Jane Greenberg and Thomas Severiens)
  – 06/2006: Chapel Hill, NC, USA (JCDL 2006)
  – 06/2007: Wien, Austria (ELPUB 2007)
  – 08/2007: Singapore (DC 2007)
Current Focus of Work

- Community is responsible for the DCMI collection of Tools and Software: http://www.dublincore.org/tools/
  - How to sort, describe, organize such a collection?
  - Collection policy?
  - What are „tools“ and „software“? What about „algorithms“ and „software packages“? E.g. should we collect OpenOffice, because it can handle DCMI MetaData?

Work in Progress:

- DCMI Tools Glossary
  http://www.dublincore.org/groups/tools/glossary.shtml

- DCMI Tools Metadata Application Profile
  http://www.dublincore.org/groups/tools/map.shtml

- DCMI Tools and Software Collection Policy
  (needs to be discussed within the Advisory Board first)
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)

Automatically Generated Metadata
Metadata generated with the aid of machine processing. See derived metadata, metadata extraction, and metadata harvesting. (Greenberg)

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)

Derived Metadata
Metadata that is automatically generated based on system programming and profiles. For example, a system program may automatically derive metadata values for "date_created", "date_modified", or resource "size". Additionally, a profile may be stored to automatically derive metadata, such as default values for "rights access", or "creator" information based on login identification. (Greenberg)

Manually generated metadata
Metadata generated by a human. Examples of classes of people generating metadata are metadata professionals (e.g., catalogers, indexers, and other persons trained to work with information standards), Web architects, content creators, and technical assistants. (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**

Metadata generated by machine processing document content. Automatic indexing and information retrieval algorithms are generally employed. For example, term frequency algorithms are used to assign subject terms. Algorithms identifying nouns and noun phrases for metadata properties (e.g., author, date, title) may be used to identify metadata element values. (Greenberg)

**Metadata Generation**

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

**Metadata Harvesting**

Automatically gathering metadata that is already associated with a resource, and which has been produced via automatic or manual means. Metadata harvested may be attached to a document (e.g., it may be encoded in the header of a Web resource), or it may be found in a metadata registry or database. (Greenberg)

**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.
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)
## DCMI Tools Metadata Application Profile

### Draft Proposal, April 23, 2007

<table>
<thead>
<tr>
<th>Namespace</th>
<th>Element</th>
<th>Qualifiers</th>
<th>Example DC-dot</th>
<th>Example Picard Tagger</th>
</tr>
</thead>
</table>
| dc        | contributor | doap:maintainer  
doa:developer  
doa:documenter  
doa:translator  
doa:tester  
(marc relator-code needs to be  
checked, if of any use here.) | Rachel Heery | developer: LukasLalinsky  
developer: RobertKaye |
| dc        | creator | (Check: same as  
dc.contributor:doap.developer  
?) | Andy Powell | |
| dc        | date | dcterms:created  
dcterms:dateCopyrighted  
dcterms:modified (is this  
implementing a new version,  
always? Needs to be clarified  
in the collection policy)  
dcterms:issued  
(give good reasoning, why we  
only offer a subset of the  
available vocabulary.) | Created: 7 July 1997 | issued: 2006-06-25 |
<p>| 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 you to specify how and where to write cleanly tagged files to your hard drive. |
| 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 (we need an example for release!) | requires: Libwww-perl, soif.pl, Jon Knight's MARC module | requires: PyQt4 Mutagen (1.7) python-musicbrainz2 isPartOf: <a href="https://musicbrainz.helixcommunity.org/release">https://musicbrainz.helixcommunity.org/release</a>: 0.7.1 |</p>
<table>
<thead>
<tr>
<th>dc</th>
<th>rightsHolder</th>
<th>(is this in dcterms? and give an example)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>dc</td>
<td>title</td>
<td>dcterms:alternative</td>
<td>DC-dot, Picard Tagger</td>
</tr>
<tr>
<td>dc</td>
<td>type</td>
<td>dcterms:dataset</td>
<td>dcterms:InteractiveResource, dcterms:software</td>
</tr>
<tr>
<td></td>
<td></td>
<td>dcterms:InteractiveResource</td>
<td></td>
</tr>
<tr>
<td>dcterms</td>
<td>audience</td>
<td>dctools:developer</td>
<td>dctools:users, dctools:developer</td>
</tr>
<tr>
<td></td>
<td></td>
<td>dctools:users</td>
<td></td>
</tr>
<tr>
<td>doap</td>
<td>location</td>
<td>(ISO-Standards on naming of location needed! Needs to be discussed, if it is really needed to be in this AP)</td>
<td>Bath, UK</td>
</tr>
<tr>
<td>doap</td>
<td>programming-language</td>
<td>(Do we have any encoding scheme? We should ask LMS-people!?)</td>
<td>Perl, Python</td>
</tr>
<tr>
<td>doap</td>
<td>operating-system</td>
<td>(Should be checked if we need a hardware-description also, e.g. for tools, made for)</td>
<td></td>
</tr>
</tbody>
</table>
mobile devices, or this is already implicated by the operating system?)

Give URI for all namespaces etc.

We also propose to use the following vocabulary for describing the tools functionalities:

- Conversion
- Crosswalk
- Metadata Creation
- Metadata Encoding
- Metadata Extraction
- Metadata Generation
- Metadata Harvesting
- Metadata Template
- Search Engine
- Translation
- Transliteration
- Validation

Highlighted text gives the results of discussion during the workshop.