Using metadata schema registry as a core function to enhance usability and reusability of metadata schemas

Mitsuharu Nagamori
University of Tsukuba, Japan
nagamori@slis.tsukuba.ac.jp

Shigeo Sugimoto
University of Tsukuba, Japan
sugimoto@slis.tsukuba.ac.jp

Abstract

Metadata schema registries have great potential to enhance usability and reusability of metadata schemas. Application profiles are a key concept for Dublin Core, and have a crucial role in promoting reuse of metadata schemas. This paper discusses basic concepts and models of metadata schemas, in order to clarify functional requirements for extending registry functions to enhance usability and reusability of metadata schemas. It also describes two experimental tools developed to help users find, browse and build metadata schemas.

Keywords: metadata schema registry; application profile; reusability of metadata schemas; metadata schema model; metadata schema interoperability.

1. Introduction

Metadata schema registries have been recognized as a crucial function for sharing authoritative information about metadata schemas (ISO/IEC JTC1 SC32 WG2, 2006). The core function of metadata schema registries is to collect, store and provide reference descriptions of metadata schemas. The Dublin Core Metadata Initiative (DCMI) is running the DCMI schema registry which currently provides the reference descriptions of the metadata terms defined in RDF Schema (DCMI, 2004). Application profiles are also a crucial component of the metadata schema registry. The IE Metadata Schema Registry by UKOLN is designed not only for metadata terms but also for application profiles (IEMSR Project, 2004).

We have been involved in the development of the DCMI schema registry since 1998 (Nagamori & Sugimoto, 2001). The registry running at the University of Tsukuba is the same software operated by DCMI (University of Tsukuba, 2005). We have been working on the registry from the viewpoint of improving the efficiency of metadata and metadata schema development, and enhancing interoperability of metadata schemas by extending the functionality of the registry. We have already reported a simple layered model designed to help users understand metadata schemas from the viewpoint of interoperability (Nagamori & Sugimoto, 2004).

Application profiles, which are widely accepted as a crucial concept for Dublin Core, are based on the “mixing and matching” metaphor (Heery & Patel, 2000). “Do not re-invent the wheel” and “Reuse existing schemas” are slogans to promote the use of application profiles. In order to promote the use of application profiles, we need suitable software tools that help users find and understand existing schemas and their components. Metadata schema registries have significant potential to serve as a hub for functions to support users. We have developed software tools to enhance the usability of metadata schemas stored in the registry. These tools include: (1) a software development support tool which semi-automatically creates software tools for metadata applications based on application profiles (Nagamori & Sugimoto, 2004); (2) a graphical browser designed for metadata vocabulary expressed in Simple Knowledge Organization System (SKOS) (Miles & Brickley, 2005) and applied to the National Diet Library Subject Headings (NDLSH) (Nagamori & Sugimoto, 2006, November); and (3) a metadata schema development support tool (Shoyama, Nagamori, & Sugimoto, 2007). These tools are designed as a functional extension of the schema registry in order to improve usability and reusability of metadata schemas.

OPEN ACCESS
This article is licensed under a **Creative Commons Attribution 4.0 International License**, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and cite the source.

https://doi.org/10.23106/dcmi.952108698

Based on these experiences, this paper aims to clarify functional requirements given to metadata schema registries to enhance usability and reusability of existing metadata schemas. First, we discuss basic models of metadata schemas from this viewpoint. Then, we describe the metadata schema development support tool and the graphical metadata vocabulary browser.

2. Metadata Schema Models and the Metadata Schema Registry

2.1. Basic Concepts of Metadata Schema

"Metadata Schema" refers to both the semantic and syntactic features of metadata. Very roughly speaking, a metadata schema is composed of a set of terms, a set of structural definitions of metadata instances, and a binding scheme for implementation. Before going into in-depth discussion, this section describes some basic concepts about metadata schemas and clarifies the meanings of the terms used in this paper.

Metadata schema components based on the model shown in our previous paper (Nagamori & Sugimoto, 2004) include:

- Property Vocabulary: A set of terms defined to express properties of a resource, e.g., title, creator, alternative and so on.
- Value Vocabulary: A set of terms which express types of property values and/or which are used as a property value, e.g. ISO-8601, DCMI Type Vocabulary, LCSH, and DDC. A value vocabulary defined as a subject heading or subject classification is called "Subject Vocabulary" in this paper.
- Implementation-Neutral Description of Metadata Structure: A set of rules which define structural constraints and features neutral to any implementation-specific description scheme, e.g. mandatory levels, repeatability/cardinality, order, and so on.
- Implementation-Dependent Description Scheme of Metadata: A set of binding rules to a specific description language, e.g., XML, HTML and RDF/XML.

The first two categories define the name(s) and meaning of a term, which give the semantic basis of the schema. We call a set of terms in these two categories of Metadata Vocabulary as a super-class of Property, Value and Subject Vocabularies. The latter two categories define syntactic features in an abstract and concrete form, respectively. In a real application environment, a set of guideline statements to create metadata instances in accordance with the application is required. An instance of description of metadata structure is called an Application Profile in this paper. FIG. 1 shows a conceptual view of an application profile. The core task in developing a metadata schema for an application is to define an application profile for the application. On top of the definition of the application profile, a physical description scheme is defined, e.g. RDF/XML. More detailed descriptions of application profiles are given by DCMI (Woodley, 2005) and CEN (CEN Workshop Agreement, 2003).

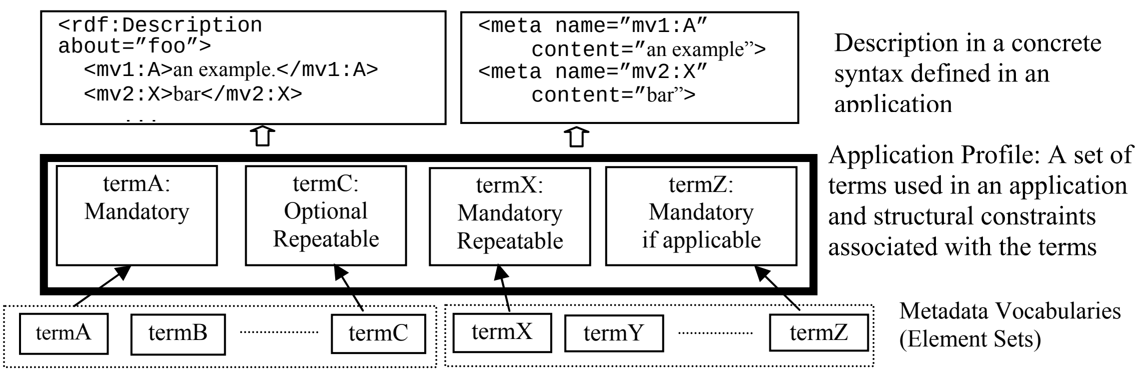

FIG. 1. Concept of Application Profile (Nagamori, 2006).

OPEN ACCESS
This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and cite the source.
https://doi.org/10.23106/dcmi.952108698

Formal specification schemes should be used to define the components in order to avoid unclear definitions. For example, RDF Schema is used to define the metadata terms by DCMI, and the JISC IEMSR project has defined a formal description scheme of application profiles based on RDF Schema (CEN/ISSS MMI-DC, 2004).

A new concept called Description Set Profile is included in the metadata schema model of Dublin Core in the recent discussion at DCMI (DCMI Architecture Working Group, 2007). A description set profile gives a set of structural constraints of a part or a whole metadata instance. We will elaborate on our model in accordance with the definition of the description set profile.

2.2. Metadata Schema Models

Conceptual modeling of metadata schemas is important in understanding the organization and fundamental features of the schemas. In this section, we describe schema models from three viewpoints – abstraction of metadata schema structure, conceptual data model for metadata schemas, and domain oriented resource modeling for metadata schemas.

(1) Abstraction of Metadata Schema Structure – Layered View of Metadata Schemas

Layered models are useful not only for identifying components and functions of metadata schemas, but also in understanding interoperability and reusability issues in metadata. For example, the DELOS white paper (Baker et al., 2003) defined a layered model of components of metadata schemas. The authors proposed a layered model which is summarized in Table 1. FIG. 1, which shows the concept of application profile, illustrates the layers – boxes in the bottom, middle and top correspond to Layers I, II and III, respectively. Thus, FIG. 1 reflects this layered model implicitly (Nagamori & Sugimoto, 2004).

TABLE 1. A layered model for metadata schemas.

Layer III	Implementation Schema: Concrete syntax to encode metadata expressed in a specific language, e.g., RDF/XML, XML, SQL, and so forth.
Layer II	Application Profiles: a set of metadata terms selected from one or more metadata vocabularies and structural constraints given to the terms
Layer I	Metadata Vocabularies: Definition of metadata terms and their sets, e.g., metadata terms defined in DCMES, LOM, and so forth.

(2) Conceptual Data Model for Metadata Schemas - Dublin Core Abstract Model

The Dublin Core Abstract Model (DCAM) defines the underlying data model for Dublin Core (Powell, Nilsson, Naeve, Johnston, & Baker, 2005). It clarifies the underlying model for resources and encoding for Dublin Core and other fundamental concepts such as Dumb-Down. From the viewpoint of the metadata schema model in this paper, DCAM primarily gives the definitions of underlying concepts (or meta-concepts) and the relationships among them. This model is crucial in formally understanding how Dublin Core is designed and organized.

(3) Resource Modeling for Metadata Schemas

Metadata schemas designed for specific domains or purposes are defined based on the data model of objects included in the domains. For example, Functional Requirements for Bibliographic Records (FRBR) has three groups of entities: for instance, group one includes Work, Expression, Manifestation and Item entities (IFLA, 1997). The data model of PREMIS (Preservation Metadata: Implementation Strategies) has five types of entities – Intellectual Entities, Objects, Rights, Events, and Agents (PREMIS Working Group, 2005). The data models for application domains are a crucial component in understanding not only a metadata schema itself but also the context of the metadata schema.

OPEN ACCESS
This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and cite the source.
https://doi.org/10.23106/dcmi.952108698

2.3. Enhancing Usability and Reusability of Metadata Schemas

(1) Supporting reuse of metadata schemas – "Do not re-invent metadata schemas"

Traditionally, a metadata schema for an application has been developed as a self-contained schema or as a subset of a comprehensive metadata schema, i.e., a schema family defined by profiles applied to a large schema. This development scheme is advantageous for achieving in-depth interoperability among metadata based on the same schema or schema family. In other words, interoperability is achieved in a community whose members use the same schema or a schema family. However, in the Internet environment, where many different communities use the same infrastructure, it is desirable to achieve metadata interoperability among different communities. On the other hand, it is also desirable to allow each community to define a metadata schema which satisfies their own requirements. These objectives are in conflict with each other.

The application profile model provides a solution to these conflicting objectives. In the application profile model, metadata terms are defined in more than one metadata element set and users are encouraged to reuse existing metadata terms rather than define new terms. On the other hand, in the conventional schema model, a schema is primarily defined to be self-contained, and terms are created anew for each new model. In the application profile model, the primary task for a metadata schema developer is to find metadata schemas and then select metadata terms that fit his/her application, or modify a metadata schema in accordance with the requirements of the application.

(2) Supporting use of metadata schemas – "Find and browse schemas online"

Users of metadata schemas – metadata creators, schema maintainers, and serious end-users – need tools to find and browse metadata terms and application profiles. Such tools are crucial for effective use and reuse of schemas. Users want to search descriptions of schemas stored in a registry. They need tools to help them grasp the overall structure of an application profile and browse the definitions of metadata terms included in the application profile. Since a metadata term has its own properties and relationships to other terms, a metadata vocabulary, which is a set of metadata terms, has a graph structure. Visual representation of metadata schemas will help users find and know metadata schemas. For example, graphical representation of a thesaurus and an application profile improves usability of metadata schemas.

(3) Enhancing understanding of metadata schema contexts – "Do not misunderstand schemas"

Metadata schemas are not independent of applications. However, descriptions of schema components stored in registries tend to lose the background and context information of the metadata schema. Information about the resource models and domain models are crucial in helping users correctly understand metadata schemas.

3. Functional Requirements to Enhance Usability and Reusability of Metadata Schemas

3.1. An Information Framework for Metadata Schema Registries

In addition to the DCMI terms, we have experimentally uploaded about 60 metadata schemas collected from the Internet into the metadata schema registry at Tsukuba (Yoshino, 2006). We have experimentally stored application profiles in the registry, and built a prototype to search and browse the application profiles (Shoyama et al., 2007).

OPEN ACCESS
This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and cite the source.
https://doi.org/10.23106/dcmi.952108698

TABLE 2. Metadata schema aspects and information for helping users.

Component Type	Components	Instantiation/Instances	Information to help users
Schema Components	Metadata Schema	• Metadata Schema as a single instance • Defined for its objectives and purposes	• General description about the metadata schema • Definition and canonical description of the metadata schema as a whole
	Application Profile	• A scheme which defines structure and structural constraints of a metadata in an application based on metadata vocabularies • Defined in a description scheme for application profiles, i.e. meta-language	• General description about an application profile • Definition and canonical description of the structure and constraints of an application profile • Description in implementation neutral schemes of an application profile
	Metadata Vocabulary	• A set of controlled terms used in a metadata description scheme, e.g., DCMES, LCSH, DDC, NDC, etc. • Defined as a property or a class	• General description about a metadata vocabulary as a set of terms • Definition and canonical description of each term of a metadata vocabulary including relationships to other terms, e.g. broader, narrower, related, super-/sub-property, etc. • Description in implementation neutral schemes of a metadata term
Schema Objectives	Objectives	• An objective of metadata schema which contains entities in the objective and aims of a metadata schema	• General description about aims and application domains of a metadata schema
	Domain/Context	• Area of application, contexts, aims of metadata description	• General description about application area, contexts, aims of metadata description
	Entity (Entities included in an underlying data model of a metadata schema)	• An entity is an objective of metadata description, e.g., entities of group 1, 2, and 3 in FRBR, intellectual entity, object, event, right, agent in PREMIS • Instance of a Class • Used as a domain and/or a range of a property	• General description about an entity included in the underlying model of the metadata schema, e.g., resource, collection, agent, rights, etc. • Definition and canonical description of an entity
Formal Description Components	Definition Scheme	• A description scheme(s) or language(s) used to encode metadata, e.g., RDF/XML, SKOS, XML, OWL, TopicMaps, etc.	• General description about a definition scheme(s) • Metadata terms expressed in an encoding scheme and associated information, e.g. namespace
	Implementation Scheme	• Implementation scheme of an application profile (e.g., XML, SQL, RDF/XML, SKOS, etc.)	• General description about implementation of a metadata schema

OPEN ACCESS
This article is licensed under a **Creative Commons Attribution 4.0 International License**, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and cite the source.

https://doi.org/10.23106/dcmi.952108698

As described in the previous section, a metadata schema has several components – the metadata vocabulary, the application profile, and an implementation scheme. In addition to information about these components, users will need information about the objectives of a metadata schema on the one hand, and a (semi-)formal definition of a metadata schema and its components on the other. Table 2 summarizes the conceptual components needed to help users use and reuse metadata schemas. Identifying these components is crucial in clarifying requirements for extending the functionality of a metadata schema registry. The first three rows show schema components, the next three rows show components related to objectives, and the last two show formal description aspects for metadata schemas. FIG. 2 illustrates relationships among the components and their sub-components.

In FIG. 2, the DCMI registry primarily covers only the blocks represented as vocabulary terms, while the JISC IEMSR registry covers the application profile blocks in addition to the vocabulary blocks. The attributes defined for schema registries are useful to convey authoritative information of the definition of terms to users. However, users will need more contextual information about the terms to help them reuse terms in their own applications (Johnston, 2005). For example, a general description of an application profile and metadata schema (as a single instance) will help users understand how an application metadata schema is organized and how and why the metadata terms are included in the application profile. Information about implementation schemes will help users design their own systems. Information about description schemes will give them information about underlying infrastructure to share the definitions of the metadata components on the network.

3.2. Functional Requirements for Metadata Schema Registries

In general, metadata schema registries provide search and browse functions of canonical descriptions of metadata terms, which is a very basic function to enhance usability and reusability of metadata schemas. The following section describes functional requirements to extend the functionality of metadata schema registries, in order to enhance usability and reusability of metadata schemas. These have been developed based on our experiences with the schema registry and also on the information demands described in the previous section.

(1) Search Across Schemas: A tool that can search across different metadata vocabularies and application profiles is required to help users find appropriate metadata terms which

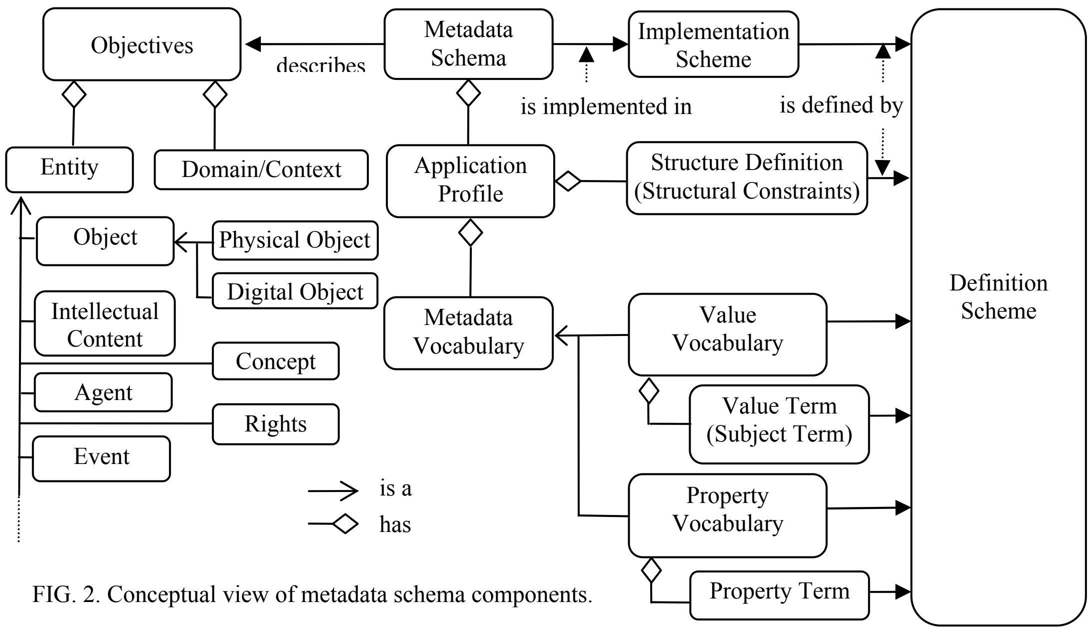

FIG. 2. Conceptual view of metadata schema components.

OPEN ACCESS
This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and cite the source.

https://doi.org/10.23106/dcmi.952108698

fit into new application profiles.

(2) Categorization of Schemas and Schema Components: Categorization of metadata terms and application profiles is crucial to help users understand the meanings of metadata terms and schemas, and to help them navigate in a large space of metadata schemas and terms. Categorization should be provided for multiple aspects, e.g. the elementary meaning of a term, an application domain, or the objectives of an area of application.

(3) Visualization of Schema and Vocabulary Structures: Visualization of metadata schemas is crucial in order to help users see the schemas from a bird's-eye view and to help them grasp the relationships among the metadata terms. A cross-schema, cross-domain graphical browsing function is crucial not only for finding metadata terms across schemas but also to understand the context of the terms.

(4) Formal Specification of Schemas: Formal descriptions of metadata schemas are indispensable not only for information sharing on the Internet but also for reusing and implementing schemas.

(5) Application Program Interfaces (APIs): Application Program Interfaces (APIs) are needed to provide both core and extended functions for other application programs.

4. Software Tools as a Functional Extension of Registries to Enhance Usability and Reusability of Metadata Schemas

Two software tools which have been designed as functional extensions of the metadata schema registry at Tsukuba are discussed here. Section 4.1 describes a tool designed to assist in application profile design. It has functions to search the metadata schema database and to help users navigate to find metadata terms. Section 4.2 outlines a graphical browser of metadata terms encoded in SKOS. This browser is named Hybrid and Network-Assisted Vocabulary Interface (HANAVI) and has been applied to NDLSH (National Diet Library Subject Headings) (Nagamori, 2006, August).

4.1. A Design Assistance Tool for Application Profiles

APdesignAssist (APdA) is a software tool to support the development process of application profiles. APdA has a metadata schema database and provides search and browsing functions. APdA uses the schema definition of application profiles based on the description scheme defined by UKOLN (CEN/ISSS MMI-DC, 2004, 2005).

APdA has the following major functions:

(1) Schema search: Text-based search function to retrieve metadata schema components expressed in RDF schema. This search includes searching by domain, range and category in addition to text-based searching. APdA has a schema directory created from the category information.

(2) Graphical schema browser: A graphical browser to show metadata terms. FIG. 3 shows a snapshot of APdA with a TouchGraph interface.

(3) Application profile builder: Interactive tool to produce an application profile in RDF/XML.

APdA has categories to show a directory of properties and to help users browse properties. Currently, APdA has 22 metadata schemas collected for the Tsukuba registry. Categories of property terms shown in Table 3 are defined and used in APdA. (A category means a class of properties.) The category term set is an open set. It will gradually grow when we add new schemas to the schema collection. As shown in the table, several Simple Dublin Core terms are used as category terms. This is a natural reflection of the basic feature of the Dublin Core.

OPEN ACCESS
This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and cite the source.

https://doi.org/10.23106/dcmi.952108698

4.2. A Graphical Browsing Tool of Subject Vocabularies based on SKOS

FIG. 4 shows a graphical representation of an NDLSH term "philosophy" displayed on HANAVI (Nagamori, 2006, August). The central node shows "philosophy" and is linked to other nodes connected by relationships BT, NT, RT and USE. In addition to the graphical representation, this system has text searching by subject term and referenced term, searching by NDC term, and "term card" display which shows the full description of a term. This interface is designed for users who maintain NDLSH and those who use NDLSH for retrieval. FIG. 5 shows a SKOS description of the NDLSH term "Language and languages -- philosophy".

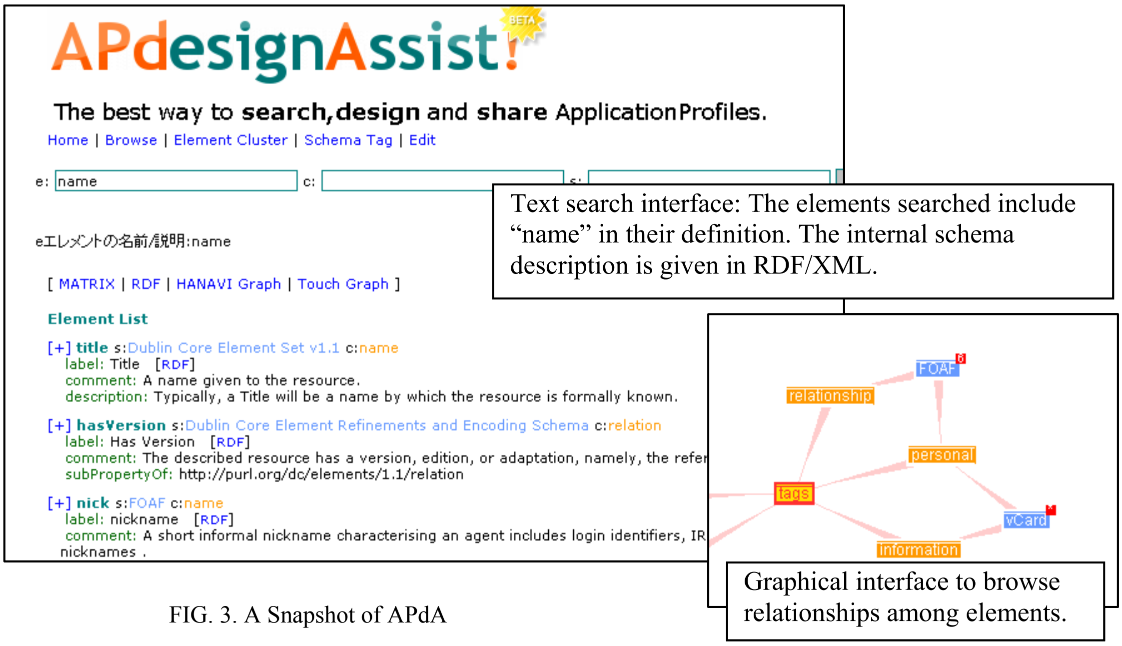

FIG. 3. A Snapshot of APdA

TABLE 3. APdA property categories.

name	contributor	Format	language	rights	Provenance
subject	Date	Identifier	relation	audience	Pedagogy
description	Type	Source	coverage	rightsHolder	

5. Discussion

5.1. Related Works

There are some services on the Internet which are useful for retrieving metadata schemas. Swoogle (UMBC ebiquity research group, 2006) is a search engine for the Semantic Web which uses RDF technologies as its basis. Swoogle helps find resources about ontologies and metadata terms. SchemaWeb (SchemaWeb, 2005) provides a directory of RDF schemas expressed in the RDF Schema, OWL and DCML+OIL schema languages. Schemas Forum worked on a registry to share information about metadata schemas (Schemas Forum, 2002).

UKOLN has been running the JISC IE Metadata Schema Registry (IEMSR) project, in which the focus is on application profiles in addition to metadata vocabularies defined in DCMES and LOM. Johnston (2005) provides an overview of the registry and discusses some important issues. He contends that contextual information associated with metadata schemas is an important issue

OPEN ACCESS
This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and cite the source.
https://doi.org/10.23106/dcmi.952108698

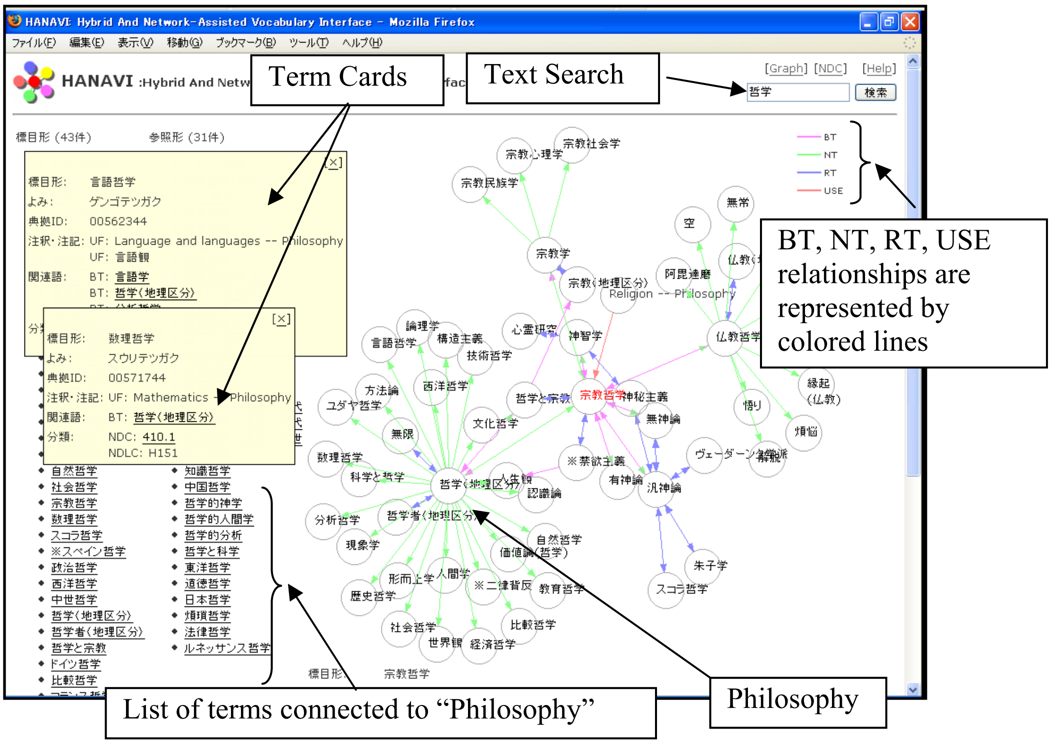

FIG 4. A snapshot of NDLSH on HANAVI.

for reuse of metadata terms in different application profiles. The CEN ISSS workshop report on Dublin Core Application Profiles presents a machine-processable representation of application profiles (CEN/ISSS MMI-DC, 2004). This report shows RDF representation for Dublin Core Application Profiles and RDF Schema description of concepts used in the RDF representation. APdA was developed based on the definition given in this report. The framework of APdA has been generalized to apply the technologies to a broader range of schemas.

In the studies presented in this paper, we focused on the basic framework necessary to enhance usability and reusability of metadata schemas. The requirements for enhancing usability of metadata schemas are needed to improve accessibility to authoritative information about metadata schemas and to provide a user-friendly interface to understand the schemas. The requirements for enhancing reusability of metadata schemas are almost the same as those for schema usability. In both cases, contextual information to help understand the background of a particular metadata schema is crucial for effective use.

5.2 Discussion and Future Work

In this paper, we described a model to help users use and reuse metadata schemas. Reusing metadata schemas is important not only for decreasing the cost of schema development but also for enhancing the interoperability of metadata schemas. We consider search and browse functions to be central components for enhancing the usability and reusability of schemas.

From our experience with the registry, we found that the set of descriptive elements included in the canonical description defined by DCMI is not sufficient for users who are searching for metadata components to use in their applications. We have not fully evaluated the sufficiency of the descriptions used to realize the search and browse functions in APdA, and will do so as part of our future work.

NDLSH on HANAVI has been used by some users at the bibliography control section of the National Diet Library (NDL) in their maintenance of NDLSH and related databases. We have not evaluated the usability of the system, but the response from users at NDL was quite positive.

OPEN ACCESS
This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and cite the source.
https://doi.org/10.23106/dcmi.952108698

```
<skos:Concept rdf:ID="http://raus.slis.tsukuba.ac.jp/ndlsh/00562344">
    <dc:identifier>00562344</dc:identifier>
    <skos:prefLabel xml:lang="ja">言語哲学</skos:prefLabel>
    <slis:annotation>
        <slis:JapaneseTransliteration>
            <slis:label xml:lang="ja">言語哲学</slis:label>
            <slis:transcription xml:lang="ja">ゲンゴテツガク</slis:transcription>
        </slis:JapaneseTransliteration>
    </slis:annotation>
    <skos:altLabel xml:lang="ja">言語観</skos:altLabel>
    <slis:annotation>
        <slis:JapaneseTransliteration>
            <slis:label xml:lang="ja">言語観</slis:label>
            <slis:transcription xml:lang="ja">ゲンゴカン</slis:transcription>
        </slis:JapaneseTransliteration>
    </slis:annotation>
    <skos:altLabel>Language and languages -- Philosophy</skos:altLabel>
    <skos:broader rdf:resource="http://raus.slis.tsukuba.ac.jp/ndlsh/00562334"/>
    <skos:broader rdf:resource="http://raus.slis.tsukuba.ac.jp/ndlsh/00572945"/>
    <skos:related rdf:resource="http://raus.slis.tsukuba.ac.jp/ndlsh/00561031"/>
    <dc:relation rdf:datatype="http://purl.org/net/tsukuba#NDC"> 801.01</dc:relation>
    <dc:relation rdf:datatype="http://purl.org/net/tsukuba#NDLC">KE13</dc:relation>
</skos:Concept>
```

FIG. 5. SKOS description of an NDLSH term.

Application of HANAVI to other vocabularies and cross-vocabulary browsing by HANAVI is one of our next goals for the graphical vocabulary browser based on SKOS.

To date, we have been developing software tools. Integration of the tools with the Tsukuba registry is still left for future work. We believe that connecting the tools will be rather straightforward using standard APIs, but collecting metadata schemas is still a fundamental problem. Collaboration is urgently required for this.

6. Concluding Remarks

We have been working on the DCMI metadata schema registry at Tsukuba since 1998. We believe that the metadata schema registry has tremendous potential to enhance usability, reusability and interoperability of metadata schemas. However, we have not realized this potential yet. More effort is needed to integrate software tools with metadata schema registries, as well as to collect and organize more schemas, so as to enhance the usability and reusability of metadata schemas and their registries.

References

Baker, Thomas, C. Blanche, D. Brickley, E. Duval, R. Heery, P. Johnston, L. Kalinichenko, H. Neuroth, and S. Sugimoto. (2003). *Principles of metadata registries* (A White Paper of DELOS Working Group of Registries). Retrieved June 26, 2007, from http://delos-noe.iei.pi.cnr.it/activities/standardizationforum/Registries.pdf.

CEN Workshop Agreement (CWA 14855). (2003). *Dublin Core application profile guidelines*. Retrieved June 26, 2007, from ftp://ftp.cenorm.be/PUBLIC/CWAs/e-Europe/MMI-DC/cwa14855-00-2003-Nov.pdf

CEN/ISSS MMI-DC WI4 Project Team. (2004). *Guidelines for machine-processable representation of Dublin Core Application Profiles, final CWA*. Retrieved June 26, 2007, from http://www.ukoln.ac.uk/metadata/cen/ws-mmi-dc/.

CEN/ISSS MMI-DC. (2005). *Schema for the CEN MMI-DC AP terms vocabulary*. Retrieved June 26, 2007, from http://www.ukoln.ac.uk/metadata/cen/ws-mmi-dc/terms/.

DCMI Architecture Working Group. (2007). *DCMI Description Set Profile*. Retrieved June 26, 2007, from http://dublincore.org/architecturewiki/DescriptionSetProfile.

DCMI. (2004). *The Dublin Core Metadata Registry*. Retrieved June 26, 2007, from http://dublincore.org/dcregistry/.

OPEN ACCESS

This article is licensed under a **Creative Commons Attribution 4.0 International License**, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and cite the source.

https://doi.org/10.23106/dcmi.952108698

Heery, Rachel, and Manjula Patel. (2000). Application profiles: Mixing and matching metadata schemas. *Ariadne, 25*. Retrieved June 26, 2007, from http://www.ariadne.ac.uk/issue25/app-profiles/.

IEMSR Project. (2004). *JISC IE Metadata Schema Registry*. Retrieved June 26, 2007, from http://www.ukoln.ac.uk/projects/iemsr/.

IFLA. (1997). *Functional Requirements for Bibliographic Records—Final report*. Retrieved June 26, 2007, from http://www.ifla.org/VII/s13/frbr/frbr.htm.

ISO/IEC JTC1 SC32 WG2. (2006). *ISO/IEC 11179, Information Technology—Metadata Registries (MDR)*. Retrieved June 26, 2007, from http://metadata-standards.org/11179/.

Johnston, Pete. (2005). What are your terms? *Ariadne, 43*. Retrieved June 26, 2007, from http://www.ariadne.ac.uk/issue43/johnston/#author1.

Miles, A. J., and D. Brickley. (2005). *SKOS Core Guide*. Retrieved June 26, 2007, from http://www.w3.org/TR/swbp-skos-core-guide/.

Nagamori, Mitsuharu, and Shigeo Sugimoto. (2001). A multilingual metadata schema registry based on RDF schema. *Proceedings of the International Conference on Dublin Core and Metadata Applications, Tokyo* (pp. 209-212).

Nagamori, Mitsuharu, and Shigeo Sugimoto. (2004). A framework of metadata schema for functional extension of metadata schema registry. *Proceedings of the International Conference on Metadata Across Languages and Cultures, Shanghai* (pp. 3-11).

Nagamori, Mitsuharu, and Shigeo Sugimoto. (2006, August). *Hybrid and network-assisted vocabulary interface*. Retrieved June 26, 2007, from http://raus.slis.tsukuba.ac.jp/.

Nagamori, Mitsuharu, and Shigeo Sugimoto. (2006, November). Representing National Diet Library Subject Headings (NDLSH) in SKOS and its graphical browser. *31st Digital Library Workshop* (pp. 13-21).

Powell, A., M. Nilsson, A. Naeve, P. Johnston, and T. Baker. (2005). *DCMI Abstract Model*. Retrieved June 26, 2007, from http://dublincore.org/documents/abstract-model/.

PREMIS Working Group. (2005). *PREMIS: Preservation Metadata Maintenance Activity*. Retrieved June 26, 2007, from http://www.loc.gov/standards/premis/.

Schemas Forum: Forum for metadata schema implementers. (2002). Retrieved June 26, 2007, from http://www.schemas-forum.org/.

SchemaWeb. (2005). Retrived June 26, 2007, from http://www.schemaweb.info/.

Shoyama, Kazuo, Mitsuharu Nagamori, and Shigeo Sugimoto. (2007). A software tool to support metadata schema development by schema search-and-reuse. *32nd Digital Library Workshop* (pp. 3-9).

UMBC Ebiquity Research Group. (2006). *Swoogle*. Retrieved June 26, 2007, from http://swoogle.umbc.edu/.

University of Tsukuba. (2005). *Tsukuba Registry*. Retrieved June 26, 2007, from http://juror.slis.tsukuba.ac.jp/.

Woodley, M. S. (2005). *DCMI glossary*. Retrieved June 26, 2007, from http://dublincore.org/documents/usageguide/glossary.shtml.

Yoshino, Mayo. (2006). *A support method for metadata schema development based on application profiles*. Tsukuba: University of Tsukuba.

OPEN ACCESS
This article is licensed under a **Creative Commons Attribution 4.0 International License**, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and cite the source.

https://doi.org/10.23106/dcmi.952108698