

SCHEMAS

Contract: N° IST-1999-10100

Forum for Metadata Schema Implementers

D74: The SCHEMAS Forum – a Retrospective Glossary

Document number:

SCHEMAS-GMD-WP7-D74-Final-20020129

| |
|----------------------------|
| General Information |
|----------------------------|

| | |
|-----------------------------------|--|
| Title | The SCHEMAS Forum – a Retrospective Glossary |
| Creator | Thomas Baker |
| Creator | Gauri Salokhe |
| Subject-Keywords | Deliverable D74; Glossary |
| Description | A “retrospective glossary” of key concepts used in the SCHEMAS Project |
| Publisher | GMD |
| Date | 2002-01-29 |
| Type | Text Manuscript |
| Format | application/msword |
| Identifier-URL | |
| Identifier-Document Number | SCHEMAS-GMD-WP7-D74-Final-20020129 |
| Language | English |
| Rights | European Commission |

Dublin Core Metadata in HTML

```
<META NAME="DC.Title" CONTENT="The SCHEMAS Forum – a Retrospective Glossary">
<LINK REL=SCHEMA.dc HREF="http://purl.org/metadata/dublin_core_elements#title">

<META NAME="DC.Creator" CONTENT="Thomas Baker">
<LINK REL=SCHEMA.dc HREF="http://purl.org/metadata/dublin_core_elements#creator">

<META NAME="DC.Creator" CONTENT="Gauri Salokhe">
<LINK REL=SCHEMA.dc HREF="http://purl.org/metadata/dublin_core_elements#creator">

<META NAME="DC.Subject" CONTENT="Deliverable D74">
<LINK REL=SCHEMA.dc HREF="http://purl.org/metadata/dublin_core_elements#subject">

<META NAME="DC.Subject" CONTENT="WP7">
<LINK REL=SCHEMA.dc HREF="http://purl.org/metadata/dublin_core_elements#subject">

<META NAME="DC.Subject" CONTENT="Glossary">
<LINK REL=SCHEMA.dc HREF="http://purl.org/metadata/dublin_core_elements#subject">

<META NAME="DC.Description" CONTENT="This document contains a Glossary.">
<LINK REL=SCHEMA.dc HREF="http://purl.org/metadata/dublin_core_elements#description">

<META NAME="DC.Publisher" CONTENT="GMD">
<LINK REL=SCHEMA.dc HREF="http://purl.org/metadata/dublin_core_elements#publisher">

<META NAME="DC.Date" CONTENT="(SCHEME=ISO8601) 2002-01-23">
<LINK REL=SCHEMA.dc HREF="http://purl.org/metadata/dublin_core_elements#date">

<META NAME="DC.Type" CONTENT="Text.Manuscript">
<LINK REL=SCHEMA.dc HREF="http://purl.org/metadata/dublin_core_elements#type">

<META NAME="DC.Format" CONTENT="(SCHEME=IMT) application/msword">
<LINK REL=SCHEMA.dc HREF="http://purl.org/metadata/dublin_core_elements#format">

<META NAME="DC.Language" CONTENT="English">
<LINK REL=SCHEMA.dc HREF="http://purl.org/metadata/dublin_core_elements#language">

<META NAME="DC.Rights" CONTENT="European Commission">
<LINK REL=SCHEMA.dc HREF="http://purl.org/metadata/dublin_core_elements#rights">
```

Dublin Core Metadata in RDF

```
<?xml version="1.0"?>
<rdf:RDF xml:lang="en"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:smes="http://www.schemas-forum.org/registry/schemas/SCHEMAS/1.0/smes#"
>

<rdf:RDF rdf:resource = "The SCHEMAS Forum – a Retrospective Glossary">
<dc:title> The SCHEMAS Forum – a Retrospective Glossary</dc:title>
<dc:creator> Thomas Baker </dc:creator>
<dc:creator> Gauri Salokhe </dc:creator>
<dc:subject> Deliverable D74 </dc:subject>
<dc:subject> Glossary </dc:subject>
<dc:subject> WP7 </dc:subject>
<dc:description> This document contains a Glossary. </dc:description>
<dc:publisher> GMD </dc:publisher>
<dc:date> 2002-01-23 </dc:date>
<dc:type> Text.Manuscript </dc:type>
<dc:format> application/msword </dc:format>
<dc:identifier> SCHEMAS-GMD-WP7-D74-Final-20020129</dc:identifier>
<dc:language> English </dc:language>
<dc:rights> European Commission <dc:rights>
</rdf:RDF>
```

| |
|--------------------------|
| Table of Contents |
|--------------------------|

| | |
|----------------------------|-----------|
| About this glossary | 6 |
| <i>Interoperability</i> | 6 |
| <i>Metadata</i> | 7 |
| <i>Model, Data</i> | 7 |
| <i>Namespace</i> | 8 |
| <i>Profile</i> | 9 |
| <i>Registry</i> | 10 |
| <i>Schema</i> | 12 |
| <i>Scheme</i> | 13 |
| <i>Semantic Web</i> | 14 |
| <i>Vocabulary</i> | 15 |
| Bibliography | 16 |

About this glossary

"SCHEMAS – A Forum for Metadata Schema Implementors", an Accompanying Measure of the EU's Information Society Technologies research programme, was designed to help implementors of information projects and services understand the diverse and often confusing landscape of new and emerging metadata standards and to use these standards effectively in creating their own metadata schemas [SCHEMAS]. Through helping implementors discover and share existing or emerging vocabularies for describing their resources, the more general goal has been to help integrate access to a diversity of resources on the Web.

Over the course of the project, however, it has become clear that the problem of metadata integration is but one example of a more general problem of semantic interoperability which manifests itself in many guises. The problem begins with the very words we use to talk about metadata vocabularies – words such as "schema" itself, which means quite different things, and evokes quite different associations, to different people.

As the two-year SCHEMAS Project draws to a close in January 2002, this "retrospective glossary" re-examines some key concepts and assumptions in the light both of project experience and of ongoing discussions on semantic interoperability in the Web community. The concepts are presented alphabetically – from interoperability and metadata to model (data model), namespace, profile, registry, schema, scheme, Semantic Web, and vocabulary – though they need not be read in this order. As this glossary makes clear, terminology in this area continues to evolve.

Interested readers will find further material and pointers to related work in the proceedings of four SCHEMAS workshops involving builders of Web resources [SCHEMAS-WORKSHOPS]; in the SCHEMAS Registry, a database of metadata vocabularies discussed below [SCHEMAS-REGISTRY]; in a series of survey reports about metadata standards and their use in trend-setting projects [SCHEMAS-WATCH and SCHEMAS-FRAMEWORK]; and in a technical paper, with numerous citations, in the Journal of Digital Information [SCHEMAS-JODI]. The work of SCHEMAS will be continued and extended in "CORES – A Forum on Shared Metadata Vocabularies", an Accompanying Measure of the Semantic Web action line scheduled to begin in May 2002.

Interoperability

In a world where a tremendous diversity of information resources have become available on a single global Web, "interoperability" has become a key buzzword. Interoperability is sometimes defined as the "ability of systems to provide services to and accept services from other systems" – ideally, in an automated manner. Systems are pictured as "talking" with each other: to "communicate" effectively, with minimum loss of information, they must "understand" each other.

In the SCHEMAS Project, and in related metadata standardisation and registry-building efforts, the emphasis is on semantic interoperability – on "speaking the same language" or on translating among different systems of meaning. As Paul Miller of the UKOLN Interoperability Focus points out, however, semantics alone are but one

aspect of information sharing. Integrating information services requires "interoperability" on several levels – from the technical (eg, common formats and protocols for representing, storing and transporting information) to the political, organisational, and cultural [MILLER]. These various aspects are so inextricably linked in particular contexts that the word "interoperability" means very little on its own.

Metadata

Broadly speaking, metadata is "structured data about other data". One classic example is the library catalog, where books are described on cards listing their Author, Title, and Subject. Since the rise of the World-Wide Web in the mid-1990s there has been considerable interest in the use of metadata for describing information on the Web. However, the prospect of merging many different types of metadata records from a diversity of sources for the purpose of integrating access to the resources they describe has cast light on linguistic aspects of semantic interoperability (see *Interoperability*; DCMI-NAMESPACE).

Metadata is a form of language, and the problem of metadata interoperability has some parallels with the problem of mutual comprehension among natural languages. Just as natural languages differ not just in their vocabularies, but in their grammars, metadata languages differ in their vocabularies as well as in their underlying data models, however ill-defined those models may often be (see *Models, Data*)

The semantic interoperability of metadata systems, at any rate, implies both shared meanings and shared grammars. As with natural language, translating one particular metadata system into the terms and grammar of another requires interpretation and may involve some loss or distortion of meaning. Recognising and accepting this inherent limit to interoperability is a hallmark of the Semantic Web philosophy (see *Semantic Web*).

Model, Data

The term "data model" refers to the "grammar" of metadata language (see *Metadata*). Within the confines of a local application, there may indeed be no particular need for a coherent data model – information providers need only share a common structure of XML tags, however arbitrarily defined, in order to be able to exchange information among themselves (see *Schema*). However, idiosyncratic local structures may not lend themselves to automated interpretation or to easy integration into larger Web portals or digital libraries.

As our experience in the SCHEMAS Project confirmed, the local metadata schemas of the world have in practice often been designed in a pragmatic, ad-hoc fashion – as fill-in-the-blank templates holding useful information but without particular regard for underlying data models. The process of translating such local models into a normalized form for the sake of interoperability is rarely entirely straightforward. To cite one very common example, the attributes of an author – eg, name, affiliation, email address, and fax number – are often nested within an author element. Such a structure can be processed by a machine that has prior knowledge of parent-child relation between, for example, Author and Fax. But it is very difficult to relate

differently nested tag structures reliably in an automated manner, and any heuristics used to do so in a particular domain are unlikely to scale up for use on the Web as a whole, where crawlers encounter a vast diversity of unknown metadata structures.

Metadata can more easily be merged when they share a common data model. The most prominent such model, Resource Description Framework, is described in the entry *Semantic Web*. In the SCHEMAS Project, the “application profile” construct was designed for expressing local schemas in RDF in order to facilitate their integration within registries and digital libraries (see *Profile*).

Namespace

The purpose of the concept “namespace” is to give metadata vocabulary terms unique identity so that multiple Web applications can reference those terms unambiguously. More specifically, the concept “namespace” is associated with a W3C Recommendation that defines an XML Namespace as “a collection of names, identified by a URI reference, which are used in XML documents as element types and attribute names” [XML-NAMESPACE]. In this context, the Uniform Resource Identifier is a means for associating an XML tag (ie, a metadata term) with a vocabulary uniquely named within a global “space” of Web names. In an RDF context, then, the Dublin Core term “extent” can unambiguously be referenced with the URI “http://purl.org/dc/terms/extent” – a handle that allows machines to merge references to the term from many different sources.

Although intended to function as unique names, URIs also look a lot like URLs – addresses of specific files on specific servers somewhere on the Web. Not unreasonably, therefore, many people expect that “clicking on” a namespace URI in a browser will call up a representation of the vocabulary referred to by that namespace, such as a Web page with authoritative definitions. As of January 2002, however, it remains unclear in the Web community whether a namespace is to be considered “just a name” or whether it also should resolve to the vocabulary named – and if so, how that vocabulary should most usefully be represented for the benefit both of human users and of machines. Indeed, there is a lack of consensus on whether namespace URIs are meant to be identifiers for terms taken individually, or whether a shared URI prefix implies that the terms belong to a common “vocabulary” (see *Schema* and *Vocabulary*).

The SCHEMAS Project, in accordance with the DESIRE Project and the Dublin Core Metadata Initiative, has used the term “namespace schema” for documents that declare the names and definitions of metadata terms. Ideally, metadata terms published with namespace URIs – and thereby made available for use by others – should be backed up by organisations or individuals responsible for their creation and maintenance. Such management authorities can range from internationally recognized standards bodies to projects or services with special data elements defined primarily for local use.

Promoting convergence among information providers on existing metadata vocabularies through referencing their namespaces was a key objective of the SCHEMAS Project. The underlying assumption has been that the number of agencies setting standards should be relatively small compared with the number of information

providers using those standards for their metadata. Rather than reinventing common metadata terms such as Title or Date, in other words, most providers should be encouraged to use appropriate terms from existing namespaces in designing their local schemas.

In order to make a clear distinction between "declaring terms" for use by others versus "reusing" those terms in information services, the SCHEMAS Project has promoted a clear distinction between "namespace schemas" (which only declare) and "application profiles" (which only reuse). As described in the entry on Semantic Web below, this distinction becomes especially useful when the namespace URIs are embedded in machine-understandable representations designed to facilitate data merging, such as Resource Description Framework (RDF) (see *Semantic Web*). Similarly, application profiles – described in the following entry – are most useful when represented in a form compatible with that of the namespace schemas [HEERY].

Profile

The notion of "application profile" used in the SCHEMAS Project traces its origins to an earlier project, DESIRE, which had noticed that information providers tend to "mix and match" terms from multiple standards in order to meet the descriptive needs of a particular project or service [HEERY]. The application profile is seen as a way to document how these projects and services make use of existing standards, adapting or constraining them for specialised purposes. By definition, an application profile cannot introduce "new" data elements; each element has to come from a particular namespace (and when a new term was needed, a namespace has to be created for it) (see *Namespace*). Optionally, a profile can annotate these elements with examples or usage notes, specifying controlled vocabularies or schemes of valid values, as needed.

The primary purpose of this style of application profile is documentary, allowing information providers to declare their metadata models in a uniform manner, which in turn allows them to be integrated into a term-level index of namespace schemas and application profiles – a registry. The documentary type of profile stands in contrast to an XML-schema type of profile, the primary purpose of which is to validate the tag structure of metadata records, and to any other type of "operational" profile directly usable within software environments for processing metadata. (See the entry *Schema* for a related discussion of "semantic schemas" versus "document schemas".)

The SCHEMAS Project took the idea of a normalised, documentary-style profile one step further by articulating a simple model for expressing the profile as a series of RDF statements. As described in more detail in a technical paper, the RDF-style profile consists of a series of statements centered around the verb "uses", as in: "Our metadata uses dc:title" or "Our metadata uses foo:email to describe a bar:agent" – where dc:, foo:, and bar: resolve to the namespace URIs of various metadata vocabularies. Indexed in a registry, these URIs serve as anchors for merging multiple references to specific metadata terms, allowing a range of useful queries for discovering which projects or services use which terms in which kinds of application contexts, together with which controlled vocabularies or value schemes [SCHEMAS-JODI].

Expressing application profiles in RDF also holds out the prospect of moving the registry out of a centralised database environment – which is neither scalable beyond a few dozen vocabularies nor sustainable as a project over the longer term – into a distributed environment where profiles are maintained by and directly harvested from the information providers themselves (see further discussion in the entry **Registry** and in SCHEMAS-JODI). Making information providers responsible for declaring their own application profiles, however, implies a pedagogical effort to clarify the purpose of an application, along with easy-to-understand guidance materials and templates that automate the creation of profiles.

Metadata schemas that have been designed from the outset with interoperability in mind will clearly be easiest to express in a profile of this type. Ideally, the process of meeting local metadata requirements with existing, standard terms will have taken place at the time a local schema is first designed. A schema designer will have known what standards are available and, if possible, how other projects and services in related areas had designed their metadata. Indeed, providing schema designers with access to information on such local usage, searchable by discipline or sector, was the primary motivation in designing the SCHEMAS Registry in the first place (see **Registry**).

In practice, however, much of the information being made available on the Web is already described with metadata that was designed before the emergence of modern metadata standards and namespace URIs, or at any rate with local application needs, rather than interoperability on the Web, foremost in mind (see **Models, Data**). The process of retrospectively mapping the terms of an existing local schema to terms in existing standards may therefore involve a process of interpretation. For example, a local XML tag called "TITLE" may be interpreted with hindsight as a Dublin Core title (<http://purl.org/dc/elements/1.1/title>).

Many standardisation communities have formalised analogous notions of "profile" for standards as diverse as Z39.50 (a protocol widely used to integrate library catalogs), IEEE Learning Objects (for describing educational materials), Digital Object Identifiers (for describing intellectual property), Dublin Core (for simple resource description), and the National Spatial Data Infrastructure in the US. All of these notions of profile aim at providing a way to extend the semantics or constrain the values of a standard in order to optimise it for a particular application, function, organisation, or user community. All of these standardisation communities, moreover, appear to face a common question of whether and how to give such profiles formal recognition. The CORES Project is planning to hold an Interoperability Forum bringing together representatives of these and other initiatives for the purpose of seeking a common approach to these issues.

Registry

The term "registry" is sometimes associated with tightly controlled network services, such as hierarchies of naming authorities for resolving persistent resource names (URNs) or formal hierarchies of agencies for maintaining the definitions of data elements, in accordance with the standard ISO/IEC 11179-6. Examples of more loosely structured, "informational" registries include a catalog of XML document specifications for reuse among service providers, maintained by the Organization for

the Advancement of Structured Information Standards (OASIS); and MetaForm, a database of metadata formats and mappings maintained at the State and University Library in Goettingen [see SCHEMAS-JODI for further references].

As used in the SCHEMAS Project, the term "registry" refers, ideally, to a database that harvests various types of metadata vocabularies from their maintainers over the Web (see *Vocabulary*). In response to queries, such a registry should provide term-level documentation of definitions and usage along with contextual annotations. It should in effect function as an indexing engine for dynamically updating, merging, and serving up a large corpus of "dictionary" entries for metadata terms. The context for such a registry is the notion of a Semantic Web where anybody or any organisation can declare a metadata vocabulary and assert a relationship between that vocabulary and any other vocabulary on the Web (see *Semantic Web*).

Our primary goal in designing such a registry has been to help implementors of information projects and services find out about metadata terms in use – both official definitions and local variations – in order to encourage harmonisation of metadata usage within particular fields and applications. The longer-term goal has been to build a corpus of machine-understandable schemas that can be accessed and processed directly by software, for example to map or convert between schemas or to configure the interface of metadata creation tools. Three SCHEMAS workshops – in Bath (May 2000), Bonn (November 2000), and Budapest (May 2001) – confirmed that project designers urgently need such a resource.

As of January 2002, however, we had only partly achieved these goals. After starting with a simple "registry" of Web links to standardisation activities and trend-setting projects, we set about implementing a database to harvest namespace schemas and application profiles expressed in RDF directly from their maintainers in an open Web environment. We based this prototype on the Extensible Open RDF (EOR) Toolkit, an open-source software development project at the Online Computer Library Center (OCLC) [EOR].

At the time, however, we found it cumbersome to work with RDF and to build user-friendly interfaces. More importantly, it became clear that good-practice conventions for declaring metadata vocabularies for use in such environments had yet to emerge. To meet the need for declaring an application profile, the SCHEMAS Project proposed a simple model for doing so in RDF, and project members participated actively in the efforts of the Dublin Core Metadata Initiative to articulate sound principles for declaring namespace schemas [SCHEMAS-JODI, DCMI-REGISTRY, DCMI-NAMESPACE; see *Namespace, Profile, Schema*]. Creating a basis for consensus on such guidelines among a broader and more diverse community of standards makers and standards users became a starting point for planning the follow-on project CORES.

In order to bring a working registry more quickly online, the SCHEMAS Project moved the registry data – vocabularies, application profiles, activity reports, and reviewer annotations – out of the RDF database environment into a centralised, relational database based on a prototype registry of an earlier project, DESIRE [DESIRE]. Both for practical reasons of database access, and in order to test the normalisation of local metadata schemas to our RDF-based model of application

profiles, we entered metadata vocabularies into the registry directly rather than harvesting them from the Web.

Although it was clear from the outset that such an effort could not scale to dozens of standards and profiles over a longer term, the resulting registry, available on the Web, provides proof-of-concept for the searching functionality and interface of a dictionary-style registry [SCHEMAS-REGISTRY]. As other, related projects were discovering in parallel, the practical possibility for creating a distributed registry environment along the lines we had envisioned will depend not just on the availability of simple tools and templates for declaring vocabularies, but on the establishment of widely accepted conventions of good practice for declaring vocabularies in a Semantic Web environment generally. Software tools for managing RDF data have significantly improved over the past year, and the follow-on CORES Project, due to start in May 2002, will provide interfaces to help non-experts declare profiles in RDF.

Declaring namespace schemas and application profiles in simple forms that are harvestable by registries is useful as a means for generating an overview of metadata vocabularies and their use in particular domains. However, the true potential of registries will only be realised when enough standardisation efforts share a common model to allow registries to serve as the basis for automated translation between vocabularies. Similarly, the idea of interoperability-oriented application profiles would come into its own if the profiles could serve not just as declarations of usage, but as templates for exporting local metadata into a form more readily mergible with metadata from other sources.

Ideally, registries will provide an environment analogous to the dictionaries needed for the stabilisation and orderly evolution of natural languages. As with natural-language dictionaries, registries should both *pre*-scribe definitions and good practice (with namespace schemas) and *de*-scribe actual metadata usage, whether sound or less-than-sound (with application profiles). Registries should also link translations of metadata vocabularies into multiple natural languages, as in the prototype registry of the Dublin Core Metadata Initiative [DCMI-REGISTRY]. By making metadata language "visible" to users, it should facilitate the identification of empirical usage trends, feeding back into a more "bottom-up" process for making standards.

Schema

In current usage, the term "schema" can refer to a wide range of things from the abstract and general to the very specific. In the abstract, "schema" is sometimes used to designate a set of semantic units (ie, metadata elements or subject headings) along with their attributes, such as name, identifier, definition, or relationship to other semantic units. In the SCHEMAS Project, we have avoided referring to such concept sets generically as "schemas" and prefer the popular term "vocabulary" (see *Vocabulary*). More narrowly, "schema" can refer to the representation of a vocabulary in a particular machine-processable form, such as an RDF or relational-database schema (a "semantic schema"). Most specifically, "schema" may refer to a file describing the tag structure of an XML-encoded document, as in an XML Document Type Definition (a "document schema").

Even more confusingly, the term "schema" is widely associated with two competing W3C specifications – XML Schema and RDF Schema [XML-SCHEMA, RDF-SCHEMA]. Broadly speaking, an XML schema is designed for parsing and validating the tag structure of metadata records (in this sense, an XML schema is a "document schema"). In contrast, an RDF schema is best at representing how particular terms relate to other terms in the schema or to terms defined in other schemas on the Web (in this sense, an RDF schema is a "semantic schema"). Developed independently by two parallel working groups in the late 1990s, the relationship between these two specifications was the cause of much confusion during the project period of SCHEMAS, and as of January 2002, the W3C was leading efforts to combine functionalities of both in an integrated schema language [RDFCore].

The SCHEMAS Project speaks generically of "vocabularies", while a semantic schema is a "namespace schema" (see *Vocabulary, Namespace*). The first SCHEMAS registry prototype was based on RDF (see *Registry*). We were often asked how XML schemas could fit into this RDF-based registry, as XML schemas also fulfilled the function we had ascribed to application profiles – of declaring of how an application has "mix-and-matched" terms from different namespaces for a specialised purpose. Our answer was two-fold: that the practical difficulties of working with angle brackets mandated a focus on one particular standard (RDF), and that the process of normalising the messy metadata models of the world to an interoperable form implies the adoption of a specific data model (again, RDF).

The style of application profile used in the SCHEMAS Project can be thought of as a "schema" inasmuch it lists a set of semantic units (metadata terms). This type of profile consists of straightforward RDF statements using a few vocabulary terms coined specifically for this purposes (see *Profile*). To be precise, however, the SCHEMAS-style profile is not a "schema" in the specific sense of the RDF Schema specification inasmuch it does not use the vocabulary described in that specification for declaring metadata terms, assigning them unique identity, or specifying class or property relationships among multiple terms. Rather, it presents a set of statements in RDF declaring that an application "uses" terms defined in RDF schemas elsewhere.

Scheme

The term "scheme" deserves a separate entry here because it is so easily confused with "schema"; indeed, "scheme" is sometimes used as a synonym for "schema". In line with the Dublin Core Metadata Initiative, the SCHEMAS Project uses the term "scheme" to designate or characterise a set of valid metadata values. For example, the scheme "ISO8601" might be used to indicate that the date value "2001-12-25" follows a string format defined in the ISO standard 8601; the scheme "LCSH" might indicate that a subject value such as the string "Trade Unions – Australia – History" is taken from a controlled vocabulary, the Library of Congress Subject Headings. The similarity between "schemes" and "schemas" is particularly confusing when talking about a controlled vocabulary such as LCSH, which may be represented as a schema, yet named with a scheme.

Semantic Web

The term "Semantic Web" is shorthand for a vision that has been articulated most eloquently by Tim Berners-Lee and is being pursued by the World Wide Web Consortium. The W3C Semantic Web Activity describes its vision as "having data on the Web defined and linked in a way that it can be used for more effective discovery, automation, integration, and reuse across various applications. The Web can reach its full potential if it becomes a place where data can be shared and processed by automated tools as well as by people." [SEMANTIC-WEB]

From a technical standpoint, the Semantic Web vision rests on a few core architectural principles: a simple, linked data model for creating webs of information about related things using metadata statements of a common pattern; the use of Uniform Resource Identifiers (URIs) and XML namespaces to give unique identity to the metadata vocabulary terms used to describe resources; and the use of XML as a universal file format. The key hypotheses underlying the Semantic Web vision are that a shared grammar is needed to ensure that humans and software will interpret metadata consistently; that clusters of simple Subject-Predicate-Object statements in the style of Resource Description Framework (RDF) can describe most of the data processed by machines; and that more complex grammars would, at any rate, not interoperate in a massively diverse Web environment. According to this approach, the Uniform Resource Identifiers used in RDF statements serve as "anchor points" for merging statements drawn or extracted from multiple sources. These anchor points may be uniquely identified Web resources or the uniquely identified metadata terms used in describing those resources.

It is recognised that the process of normalising the diversity of metadata constructs of the world to a simple, uniform, almost pidgin-like statement grammar may involve a certain loss of specificity, and that exporting statements to unintended contexts may not always make sense, but these problems are accepted as an inevitable aspect of imperfect communication in an imperfect world. Rather, the more modest goal is "partial understanding" – the lossy and selective merging of data from underlying models that are semantically and structurally richer and more diverse. To Tim Berners-Lee, the imperfect nature of this understanding is an inevitable limitation to the prospect of sharing data between programs and resources that have been designed independently. As he likes to point out, the chaotic Web we have today has plenty of broken links, yet it continues to grow exponentially. By analogy, he suggests that the inconsistency and errors involved in massively re-purposing data for unintended uses will be more than offset by the benefits of integrating access to resources and data.

This notion of merging access to a diversity of resources through normalised metadata statements in RDF has been a guiding idea for the SCHEMAS Project. The metadata of the world follows a huge diversity of incompatible data models, and merging this metadata on any large scale inevitably requires translation into a common grammar. In this sense, if RDF did not exist, something like it would need to be invented.

Creating coherence, or "making sense", of metadata on a grand scale must involve an imperfect process of translation, even simplification. Instead of asking machines to understand people's language, Tim Berners-Lee has suggested that the vision of a Semantic Web involves asking people to make the extra effort to speak, as it were,

machine-understandably. The hypothesis underlying the SCHEMAS style of application profile is that the extra effort taken by information and service providers to normalise a view of their metadata will reap rewards in the form of better integration of access to resources (see *Application Profile*).

The problem of mapping diverse conceptual structures among themselves, however, is not unique to the metadata community. The same problem is being encountered for mapping databases, thesauri, and ontologies. Ontologies, for example, may be expressed in different representational languages or data models, or on the basis of ill-defined or inconsistent conceptual models that may be pragmatically or locally useful, but difficult to map automatically. From their various perspectives, these diverse communities seem to be reaching the same conclusion: that the mapping process can be automated only in part, and that manual intervention by experts is usually needed to complete (or correct) the job. This implies the more general conclusion that the large-scale merging of metadata cannot reliably be left entirely to algorithms and heuristics, but would benefit from mapping constructs that have been vetted by humans – such as application profiles of the type discussed here.

Vocabulary

In 1999, we named our project SCHEMAS because we thought the term best described the object of our interest. As the entry for "schemas" in this glossary shows, however, the term turned out to be more problematic than we had imagined. In accordance with a more general trend, we now prefer to speak generically of metadata "vocabularies". In our usage, the term evokes a semantically rich dictionary environment, with pointers to related terms – more than just a flat word list. (Another common synonym for “vocabulary” is “element set”. Similarly, though we prefer to speak of metadata "terms", the term "elements" is a close synonym.) It is not clear to us whether it is useful to think of an application profile, in the sense described above (ie, as a declaration of usage), as a “vocabulary” (see *Profile*).

Although "vocabulary" seems to be the preferred term in contexts as diverse as Dublin Core Metadata Initiative, W3C, and INDECS, there remains some ambiguity for people who associate vocabularies with controlled lists of metadata values, such as subject headings, or with other types of controlled vocabularies such as thesauri and ontologies. This ambiguity, however, may be unavoidable, as it is unclear how metadata vocabularies really differ, if indeed at all, from any other type of controlled vocabulary.

There are parallel efforts in the world of thesauri and ontologies to build registry environments that harvest and link those vocabularies in machine-processable ways. Ideally, all of these various vocabularies, metadata included, would be declared using namespace URIs and in ways that are reusable by a broad range of registries and application environments. As of January 2002, the most likely candidate for such a common format is RDF, though a broad consensus on good-practice guidelines for declaring namespaces and managing vocabularies over time has yet to emerge.

Bibliography

[DCMI-NAMESPACE] <http://dublincore.org/documents/2001/09/17/dcmi-namespace/>

[DCMI-REGISTRY] <http://www.dublincore.org/groups/registry>

[DESIRE] <http://desire.ukoln.ac.uk/registry/>

[EOR] <http://eor.dublincore.org>

[HEERY] Rachel Heery and Manjula Patel, Application profiles: mixing and matching metadata schemas, Ariadne 25, September 2000, <http://www.ariadne.ac.uk/issue25/app-profiles/intro.html>.

[MILLER] Paul Miller, "Interoperability What is it and Why should I want it?", 21 June 2000, Ariadne Issue 24, <http://www.ariadne.ac.uk/issue24/interoperability/>.

[RDF-SCHEMA] <http://www.w3.org/TR/rdf-schema/>

[RDFCore] <http://www.w3.org/2001/sw/RDFCore/>

[SCHEMAS] <http://www.schemas-forum.org/>

[SCHEMAS-FRAMEWORK] <http://www.schemas-forum.org/stds-framework/>

[SCHEMAS-JODI] <http://jodi.ecs.soton.ac.uk/Articles/v02/i02/Baker/>.

[SCHEMAS-WATCH] <http://www.schemas-forum.org/metadata-watch/>

[SCHEMAS-WORKSHOPS] <http://www.schemas-forum.org/workshops/>

[SEMANTIC-WEB] <http://www.w3.org/2001/sw/Activity>

[XML-NAMESPACE] <http://www.w3.org/TR/REC-xml-names/>

[XML-SCHEMA] <http://www.w3.org/TR/xmlschema-0/>