Describing Namespaces with GRDDL

Erik Wilde, ETH Zürich (Swiss Federal Institute of Technology)

ABSTRACT

Describing XML Namespaces is an open issue for many users of XML technologies, and even though namespaces are one of the foundations of XML, there is no generally accepted and widely used format for namespace descriptions. We present a framework for describing namespaces based on GRDDL using a controlled vocabulary. Using this framework, namespace descriptions can be easily generated, harvested and published in human- or machine-readable form.

Categories and Subject Descriptors

H.3.2 [**Information Storage and Retrieval**]: Information Storage—*File Organization*

General Terms

Management, Languages

1. DESCRIBING SCHEMAS

While an XML Schema describes constraints for a class of XML documents, but it does not convey any information about the semantics. For most application scenarios, the semantics of a schema are described in some informal way, in most cases using simple prose. It would be useful to have some standard mechanism how to associate this human-readable description with the XML Schema in a standard way. There is no established way for doing this, and there are two main approaches to solve this problem: (1) It is possible to embed additional information within the schema. (2) Since the `targetNamespace` of a schema defines a namespace name according to the *XML Namespaces* [1] recommendation, it is possible to associate additional information for a schema through the namespace name.

While both approaches solve the problem, we chose to adapt the second approach, because it better separates the schema implementation from the schema description. Schema descriptions as described here are generic enough to not only serve as XML Schema descriptions, but as descriptions for any vocabulary associated with a namespace name. Therefore, we subsequently refer to them as *namespace descriptions*. The Web architecture [3] does not define a data format for namespace descriptions, but recommends that namespace names should point to some kind of description.

2. NAMESPACE DESCRIPTIONS

Despite the fact that the Namespaces specification itself does not require any resource to be available at a namespace's URI, it is convenient if this is the case. The approach of having HTML pages serving as namespace descriptions is useful for humans, but makes it very hard to process this information automatically. In many cases, it would be beneficial to have machine-readable namespace descriptions, because these could be collected and compiled into a database of namespaces and related resources.

In an effort to create a namespace description language to combine the human-readability of HTML documents which machine-readable semantics, the *Resource Directory Description Language (RDDL)* was invented. RDDL used the unpopular XLink standard, and thus the *Gleaning Resource Descriptions from Dialects of Languages (GRDDL)* [2] language was developed, which defines a way how to expose the machine-readable information as *RDF* [4].

The GRDDL model assumes that the machine readable information of an XHTML page is extracted by using an XSLT program that transforms the GRDDL into RDF statements. Even though this adds little to the expressiveness of RDDL, it is a little bit easier to handle (because the machine-readable information can be encoded at the users discretion, as long as it can be transformed into RDF), and may gain more popularity because it uses RDF rather than XLink.

A namespace description is a set of machine-readable information about how other resources are related to the namespace, such as schemas defining the namespace's vocabulary, or documentation for the namespace application. In order to make GRDDL work, this vocabulary of how linked resources related to the namespace must be well-known.

2.1 Description Roles

GRDDL namespace descriptions serve as a supplement to the documentation, providing machine-readable documentation that can be used to compile a directory of schemas. To make this directory as rich as possible, a number of roles that must or can be used to describe schemas has to be defined. These roles describe how a particular resource relates to the namespace being described.

The description roles thus constitute that fraction of namespace description that should be available in a machine-readable way, so that it can be collected and processed. Technically, the resource roles are defined in a simple XML document, which serves as configuration for defining the GRDDL namespace description format described in the following section, and for the RDF Schema used for the information extraction process described in Section 3.

2.2 Creating Descriptions

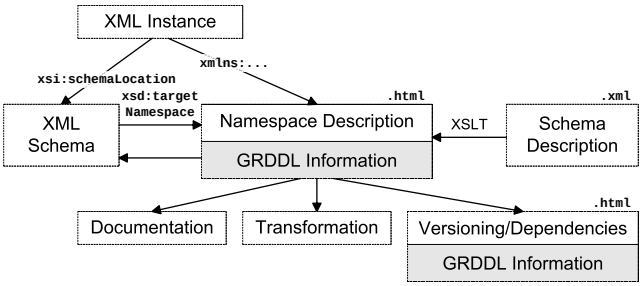
Schema authors must create GRDDL descriptions for their schemas, and they must follow the guidelines requiring certain kinds of roles to be present. However, schema authors are free how they create the GRDDL. While many choose to write their GRDDL by hand, others don't want to "learn" GRDDL, even though it is very easy to learn.

For these users, we provide a simple XML Schema that

WWW 2005, May 10–14, 2005, Chiba, Japan.
ACM 1-59593-051-5/05/0005.

can be used to capture all the information required for a GRDDL document, and an *XSL Transformations (XSLT)* program to generate the GRDDL that then serves as namespace description. This schema for namespace descriptions is generated from the list of resource roles described in the previous section.

A second schema — also generated from the list of resource roles described in the previous section — is required for describing the attributes (this schema does not define any elements) that convey the machine-readable information within the GRDDL document. Some of the attributes are defined to appear on XHTML `a` elements, augmenting the link with well-defined semantics. Other attributes appear on XHTML `div` elements and are used to enclose textual descriptions. Using these attributes, namespace descriptions contain all the information that we want to make accessible in machine-readable form.



This figure shows the complete model of how namespace descriptions are used. We assume that the namespace description is generated, so that the actual GRDDL document located at the namespace URI of the schema is an XHTML document generated by XSLT. The generated XHTML contains information about associated resources (the schema described, documentation, and transformations), and some of these associated resources may be namespace descriptions themselves (versioning and other dependencies).

3. HARVESTING DESCRIPTIONS

As pointed out in Section 2, namespace descriptions are GRDDL documents, which means they are XHTML with embedded semantic information. This makes automated processing of namespace descriptions easy, because only the semantic information has to be extracted. This task is best done by XSLT, which means that harvesting namespace descriptions means collecting GRDDL documents, and then using XSLT to generate RDF from these documents.

As pointed out in Section 2.2, we do not require schema authors to generate their namespace description. They can generate it, in which case the schema and the XSLT for the generation will guarantee an error-free namespace description. However, if the namespace description is generated by hand, errors may be introduced, so that the harvesting also needs to validate the harvested documents.

Validity in our context means that the harvested descriptions must be valid GRDDL, and that they satisfy all requirements for namespace descriptions as detailed in Section 2.1. If harvested descriptions are invalid, they are excluded from further processing stages and the description originator is contacted, if possible. This does not interfere with the overall process of harvesting and subsequently publishing all valid namespace descriptions.

Validation (as well as RDF generation) in the current implementation is done using an *XSLT 2.0* program. The reason for this is that XML Schema does not provide a reasonable way of validating a schema that is tightly integrated with a host language, and that XSLT 1.0 does not provide any support for checking datatypes. The result of the XSLT-based validation is a report containing a list of warnings or errors raised during the validation process.

4. PUBLISHING DESCRIPTIONS

After the harvesting and validation process, GRDDL documents are processed using XSLT, which after joining the individual RDF graphs results in a single RDF graph describing all harvested namespace descriptions. This aggregated set of descriptions is published as XML and XHTML.

XHTML is published as heavily crosslinked pages, enabling users to retrieve all the information present in the RDF using a regular browser. Using search features, it is possible to search for specific text in all literal information, so that access through the XHTML pages is provided through search-based retrieval as well.

For users interested in a machine-readable description of the collected data, the data is published as XML. This XML uses more traditional structures using standard ID/IDREF references rather than being based on RDF. Even though GRDDL uses an RDF-based data model, it was decided that an application-specific XML Schema is better suited for representing the namespace descriptions. The reason for this is that RDF is not well-suited for processing it with XPath/XSLT, whereas a suitably designed XML can be processed very simply by users with relatively little XPath or XSLT experience.

5. CONCLUSIONS

Our namespace description approach implements the idea of a *light-weight Semantic Web*, searching for the middle ground between the considerable effort necessary to create machine-readable descriptions for many details of an IT environment, and the absence of any machine-readable description in the plain namespace handling defined by the recommendation.

In an effort to implement the light-weight Semantic Web as easily, standards-compliant and future-proof as possible, we employed a variety of Web technologies such as XML, XML Schema, XSLT, GRDDL, and RDF. Using these technologies and combining them in the most efficient way enabled us to implement what we consider to be a gap in the XML landscape of today without too much effort.

6. REFERENCES

[1] Tim Bray, Dave Hollander, and Andrew Layman. Namespaces in XML. World Wide Web Consortium, Recommendation REC-xml-names-19990114, January 1999.

[2] Dominique Hazaël-Massieux and Dan Connolly. Gleaning Resource Descriptions from Dialects of Languages (GRDDL). World Wide Web Consortium, W3C Coordination Group Note NOTE-grddl-20040413, April 2004.

[3] Ian Jacobs and Norman Walsh. Architecture of the World Wide Web, Volume One. World Wide Web Consortium, Recommendation REC-webarch-20041215, December 2004.

[4] Ora Lassila and Ralph R. Swick. Resource Description Framework (RDF) Model and Syntax Specification. World Wide Web Consortium, Recommendation REC-rdf-syntax-19990222, February 1999.