

Structuring Namespace Descriptions

Erik Wilde
ETH Zürich

ABSTRACT

Namespaces are a central building block of XML technologies today, they provide the identification mechanism for many XML-related vocabularies. Despite their ubiquity, there is no established mechanism for describing namespaces, and in particular for describing the dependencies of namespaces. We propose a simple model for describing namespaces and their dependencies. Using these descriptions, it is possible to compile directories of namespaces providing searchable and browsable namespace descriptions.

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

General Terms: Management, Languages

1. INTRODUCTION

XML Namespaces [2] were introduced to identify vocabularies within XML documents. XML Schema extended this to also include types and other schema components. The XML Namespaces recommendation does not prescribe any internal structure of a namespace, so it is entirely up to the maintainer of a namespace to decide whether such a structure exists and if so, what this structure looks like.

XML Namespaces are in heavy use throughout all XML technologies, but in virtually all scenarios namespace names are treated as tags only, which means they are just compared for equality to check whether some name is in a given namespace. The W3C's Web architecture [4] states that a standardized format for namespace descriptions (often called *namespace documents*) would be useful, but so far little work has been done in this area.

For small and self-contained projects or scenarios, namespace documents may provide a convenient way to provide access to information associated with a namespace, but in most cases this information is available and well-known anyway. In larger and more complex scenarios without central coordination, namespace documents provide a much more needed service, because they serve as the hub for information about namespaces. By collecting and publishing namespace documents in such a scenario, it is then possible to provide a directory of namespace descriptions.

2. DESCRIPTIONS

Describing namespaces, which means making machine-readable information available at a namespace's URI, can be

done in different ways. The two most popular approaches so far use different ways for doing it. *RDDL* [1] relies on embedding the information in the namespace document directly, using a well-defined format for identifying the machine-readable information. *GRDDL* [3] uses an approach where the description can use any format and alternatively can be obtained in RDF by applying a transformation to the description document.

Our approach uses a way which technically is GRDDL-compliant, but is easier to implement. The document found at the namespace URI is a human-readable XHTML and contains a link (in the HTML's head) to the XML containing the machine-readable description. The XML is based on an XML Schema which is the authoritative format for our namespace descriptions.

A description contains a number of *facets*, each of them representing different concepts of namespace-related information. These facets include a rather small number of concepts, ranging from a namespace's title, a short descriptive text, and the preferred prefix, to links to resources such as tools, examples, and human-readable documentation. Some of the facets of a description constitute associations between namespaces, and these facets are described in greater detail in the following section.

3. DESCRIPTION ASSOCIATIONS

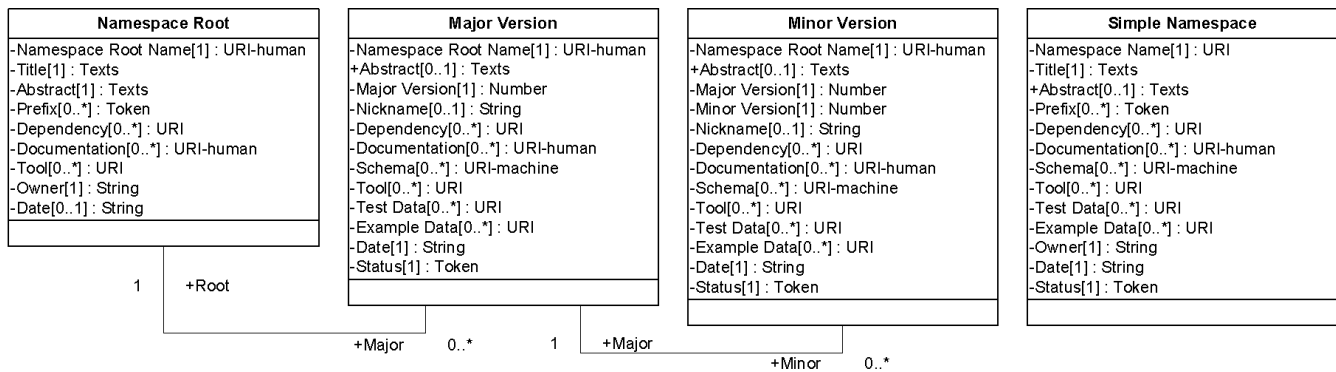
Namespaces facets constitute associations with other resources, which may or may not be identified by a namespace name. If they have a namespace name associated with them, we recommend to use this name to represent the association rather than a less useful identification (such as the URI of a schema document's location). This is only a recommendation, since it is not necessary that resources associated with a namespace are identified by a namespace name.

There is one special case of namespace associations which is treated especially, however, because this affects the organization of namespace descriptions themselves, and this is the versioning of namespace descriptions.

3.1 Namespace Versioning

There is no established standard for versioning namespaces, i.e. the question of when and how namespace names should be changed for different versions of a vocabulary. In our context, versioning is based on three levels, but only two of these levels include namespace names:

1. *Namespace Root:* This namespace is not the namespace of a concrete vocabulary, it only serves as the root for concrete major versions of a vocabulary. Thus, this namespace contains only the names of other namespaces.



2. *Major Version*: If a vocabulary uses minor versions, the major version namespace contains the names of the minor versions, as well as the vocabulary itself. If the vocabulary only uses major versions, the major version namespace name is the identification of exactly one major version vocabulary. The namespace name of the major version namespace is the concatenation of the root namespace's name, a slash character, and the major version number.
3. *Minor Version*: A vocabulary's minor version is identified by the namespace name (which only contains the major version number) and a minor version number. This number is part of the namespace description, and it is also represented in schemas and instances using dedicated attributes.

The above versioning method and structure (shown in Figure 3.1) only applies to namespaces defined in our domain. External namespaces (such as standard namespaces for XML and XML Schema) do not use this structure, and they can be described using a fourth class of descriptions, which we call *Simple Namespaces*.

All descriptions use a similar set of description facets (as described in Section 2), but depending on the concrete namespace's class, a different subset of these facets is available.

4. NAMESPACE DIRECTORY

By collecting namespace descriptions, it is possible to compile a directory of namespace descriptions, providing easy access to everybody searching information about any of these namespaces. The directory contains namespaces from within our domain (using the versioning structure) as well as outside namespaces (being described as simple namespaces), and by providing descriptions for both of these classes of namespaces, it is possible to compile a comprehensive and interlinked set of descriptions.

5. IMPLEMENTATION

We have applied the model proposed here to the area of e-Government in Switzerland, where a large number of XML-related vocabularies exist, some of them defined by various Swiss authorities, some of them adopted from standardization organizations or other bodies. This initial collection of namespaces has been described, and this set of descriptions has been augmented with descriptions of some of the core

namespaces of XML-related standards. This set of namespace descriptions has been compiled into a directory of Swiss e-government namespaces.

The main goal behind this directory is to provide a common access point for people searching information about certain namespaces. Since the namespace descriptions also provide access to additional data (such as documentation, tools, and test data), this directory serves as a unifying platform for collecting and distributing information about XML-related activities in Switzerland.

The service has only been established recently, but the initial feedback has been very positive, because in the opinion of many XML developers, namespace names should point to information about the namespace, and by using the structure described here, we can provide information about the namespace and even about other namespaces and resources which are associated with it.

6. CONCLUSIONS

Using the namespace description format and structure presented here, it is possible to provide a uniform and rich set of descriptions about namespaces in an XML-based environment. While the exact versioning strategy and method may be different in other environments, the basic concept of defining and representing associations between namespaces is applicable to many different settings. The format has been defined to be easily usable and understandable, so that namespace maintainers do not perceive it as a big additional burden they rather avoid. This design should improve the ratio between the effort required to describe a namespace, and the benefit for all users of this namespace.

7. REFERENCES

- [1] JONATHAN BORDEN and TIM BRAY. Resource Directory Description Language (RDDL) 2.0, January 2004.
- [2] TIM BRAY, DAVE HOLLANDER, and ANDREW LAYMAN. Namespaces in XML. World Wide Web Consortium, Recommendation REC-xml-names-19990114, January 1999.
- [3] DOMINIQUE HAZAËL-MASSIEUX and DAN CONNOLLY. Gleaning Resource Descriptions from Dialects of Languages (GRDDL). World Wide Web Consortium, W3C Team Submission SUBM-grddl-20050516, May 2005.
- [4] IAN JACOBS and NORMAN WALSH. Architecture of the World Wide Web, Volume One. World Wide Web Consortium, Recommendation REC-webarch-20041215, December 2004.