Network Working Group
Request for Comments: 3151
Category: Informational

N. Walsh
Sun Microsystems, Inc.
J. Cowan
Reuters Health Information
P. Grosso
Arbortext, Inc.

August 2001

A URN Namespace for Public Identifiers

Status of this Memo

Copyright Notice

Abstract

This document describes a URN (Uniform Resource Name) namespace that is designed to allow Public Identifiers to be expressed in URI (Uniform Resource Identifiers) syntax.

1. Introduction

XML [1] external entities have two identifiers: a system identifier and a public identifier. The system identifier is a URI, by definition, but the public identifier is simply a string.

Historically, the system identifier of an external entity has been a local, or system-specific identifier while the public identifier has been a more global, persistent name.

Unfortunately, public identifiers do not fit neatly into the existing web architecture because they are not legal URIs. Many new specifications (XSLT, XML Schema, etc.) have the implicit or explicit requirement that all external identifiers be URIs.

The purpose of this namespace is to allow public identifiers to be encoded in URNs in a reliable, comparable way.

This document describes a scheme for representing public identifiers as URNs by introducing a public identifier namespace, "publicid".

This namespace specification is for a formal namespace.

1.1 Public Identifiers

Any string which consists only of the public identifier characters (defined by Production 13 of Extensible Markup Language (XML) 1.0 Second Edition [1]) is a legal public identifier.

In addition to the character set restriction, public identifiers must be normalized by changing all strings of whitespace (the characters #x20, #x9, #xD, and #xA) to single space characters (#x20), and removing all leading and trailing whitespace.

In keeping with this specification's goal of allowing public identifiers to be encoded in a reliable, comparable way, this specification mandates that public identifiers be normalized before encoding them into URNs. Throughout this specification, we assume that normalization has already been performed.

1.2 Formal Public Identifiers

SGML [2] defines a restricted subset of public identifier called a "Formal Public Identifier" (FPI).

FPIs are strings composed from the same range of characters as public identifiers, but with an explicit internal structure. The structure of Formal Public Identifiers is normatively described in SGML [2]; we review it here for convenience.

Most Formal Public Identifiers consist of the following fields, in this order: an owner identifier, a public text class, a public text description, a public text language or public text designating sequence, and an optional public text display version.

Owner identifiers may begin with "-//" or "+//"; otherwise "//" is used to delimit fields in the FPI (with the exception of the public text class which is delimited from the public text description by a space).

In other words, most FPIs look like this:

```
owner//class description//language//version
```

and most owners begin with "+//" or "-//", although they are not required to. Here are some example FPIs:

+//IDN python.org//DTD XML Bookmark Exchange Language 1.0//EN//XML
-//OASIS//DTD DocBook XML V4.1.2//EN
-//ArborText::prod//DTD Help Navigation Document::19970708//EN
ISO/IEC 10179:1996//DTD DSSSL Architecture//EN
ISO 8879:1986//ENTITIES Added Latin 1//EN

This document describes an algorithm for encoding public identifiers into URNs that explicitly allows the structured nature of formal public identifiers to be preserved. However, an algorithm for correctly identifying a Formal Public Identifier and determining the various fields within it is out of scope for this document and not necessary for the implementation of this URN namespace.

2. Specification Template

Namespace ID:

"publicid" requested.

Registration Information:

Registration Version Number: 1
Registration Date: 2001-05-08

Declared registrant of the namespace:

Norman Walsh
Sun Microsystems, Inc.
One Network Drive MS UBURO2-201
Burlington, MA
01803-0902

Norman.Walsh@East.Sun.COM

Declaration of structure:

The Namespace Specific String (NSS) for URNs in the "publicid" namespace has the following structure:

urn:publicid:{transcribed-public-identifier}

Where:

{transcribed-public-identifier} is the text of the public identifier transcribed according to the following rules:

- A space in the public identifier is transcribed as "+". Whitespace normalization must be performed before constructing a URN in the "publicid" namespace, therefore adjacent "+" characters never occur in URNs in this namespace.
- The sequence of characters "//" is transcribed as ":".
- The sequence of characters "::" is transcribed as ";".
- A literal "+" character is transcribed as "%2B".
- A literal ":" character (except in "::") is transcribed as "%3A".
- A literal "/" character (except in "//") is transcribed as "%2F".
- A literal ";" character is transcribed as "%3B".
- A literal "'" character is transcribed as "%27".
- A literal "?" character is transcribed as "%3F".
- A literal "#" character is transcribed as "%23".
- A literal "%" character is transcribed as "%25".

The special rules for "//" and "::" are designed to preserve the structured nature of formal public identifiers without requiring the translator to have special knowledge of FPI syntax.

The rules for "+", ":", "/", and ";" are required to preserve literal occurrences of these characters in the 'publicid' URN namespace.

The remaining characters, " " (space), "'", "?", "#", and "%", are the only other legal characters in public identifiers that cannot be literally transcribed into a URN by the rules of RFC 2141 [4] and RFC 2396 [5].

Relevant ancillary documentation:

Extensible Markup Language (XML) Version 1.0 Second Edition [1]
Standard Generalized Markup Language (SGML) [2]
Registration procedures for public text owner identifiers [3]

Identifier uniqueness considerations:

The identifier uniqueness considerations for URNs in the "publicid" namespace are the same as the identifier uniqueness considerations for public identifiers. Formal Public Identifiers with registered owner identifiers are required to be unique. For unregistered owner identifiers and informal public identifiers, they may or may not be unique. No enforcement policy can be asserted.

Identifier persistence considerations:

The persistence of URNs in the "publicid" namespace is the same as the persistence of the corresponding public identifier.

The "publicid" namespace is available for a wide range of uses; it cannot be subjected to a uniform persistence policy. As a general rule, formal public identifiers with registered owner identifiers are more likely to be persistent than informal public identifiers or formal public identifiers with unregistered owner identifiers.

One exception to this rule is the "IDN" scheme for producing a registered owner identifier from a domain name. That scheme contains at least all the weaknesses associated with the persistence of domain names.

It is important to note that a properly registered owner identifier can apply any policy desired to the portion of the "publicid" URN namespace identified by that owner identifier.

Process of identifier assignment:

Identifiers in the "publicid" namespace are assigned by applying the conversions described above to a public identifier. In order to provide a URN in this namespace for a resource that does not have a public identifier, one must be created (according to the rules for creating public identifiers).

There is no requirement that a resource have only one public identifier.

Process of identifier resolution:

Identifiers in the "publicid" namespace may be resolved by the same policies and procedures as public identifiers. Public identifiers can be resolved in many different ways. Many existing systems provide facilities for resolving them by way of OASIS TR9401 [6] Catalog files. Other systems resolve them by mapping each component to a local pathname component. And some systems simply "know about" a fixed set of public identifiers. In addition, URNs in the ’publicid’ namespace may be resolvable by other mechanisms unique to URIs (such as caches).

Rules for Lexical Equivalence:

Whitespace normalization is performed before constructing a URN in the "publicid" namespace, so URNs are lexically equivalent if and only if they are lexically identical.

Conformance with URN Syntax:

No special considerations. URNs in this namespace conform to both RFC 2141 and RFC 2396.

Validation mechanism:

None specified.

Scope:

Global

3. Examples

The following examples are not guaranteed to be real. They are listed for pedagogical reasons only.

"ISO/IEC 10179:1996//DTD DSSSL Architecture//EN" becomes
"urn:publicid:ISO%2FIEC+10179%3A1996:DTD+DSSSL+Architecture:EN"

"ISO 8879:1986//ENTITIES Added Latin 1//EN" becomes
"urn:publicid:ISO+8879%3A1986:ENTITIES+Added+Latin+1:EN"

"-//OASIS//DTD DocBook XML V4.1.2//EN" becomes
"urn:publicid:-:OASIS:DTD+DocBook+XML+V4.1.2:EN"

"+//IDN example.org//DTD XML Bookmarks 1.0//EN//XML" becomes
"urn:publicid:%2B:IDN+example.org:DTD+XML+Bookmarks+1.0:EN:XML"

"-//ArborText::prod//DTD Help Document::19970708//EN" becomes
"urn:publicid:-:ArborText;prod:DTD+Help+Document;19970708:EN"

"foo" becomes
"urn:publicid:foo"

"3+3=6" becomes
"urn:publicid:3%2B3=6"

"-//Acme, Inc.//DTD Book Version 1.0" becomes
"urn:publicid:-:Acme,+Inc.:DTD+Book+Version+1.0"

4. Security Considerations

There are no additional security considerations other than those normally associated with the use and resolution of URNs in general.

References

[1] W3C, XML WG, "Extensible Markup Language (XML) 1.0 Second Edition", February 1998, <http://www.w3.org/TR/REC-xml>.

[2] JTC 1, SC 34, "ISO 8879:1986 Information processing -- Text and office systems -- Standard Generalized Markup Language (SGML)", 1986.

[3] JTC 1, SC 34, "ISO/IEC 9070:1991 Information technology -- SGML support facilities -- Registration procedures for public text owner identifiers", 1991.

[4] Moats, R., "URN Syntax", RFC 2141, May 1997.

[5] Berners-Lee, T., Fielding, R. and L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax", RFC 2396, August 1998.

[6] Grosso, P., "Entity Management: OASIS Technical Resolution 9401:1997 (Amendment 2 to TR 9401)", Sep 1997, <http://www.oasis-open.org/html/tr9401.html>.

Authors' Addresses

Norman Walsh
Sun Microsystems, Inc.
One Network Drive MS UBURO2-201
Burlington, MA 01803-0902
US

EMail: Norman.Walsh@East.Sun.COM

John Cowan
Reuters Health Information
45 West 36th St, 12th Floor
New York, NY 10018
US

EMail: jcowan@reutershealth.com

Paul Grosso
Arbortext, Inc.
1000 Victors Way
Ann Arbor, MI 48108-2744
US

EMail: pgrosso@arbortext.com

Full Copyright Statement

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.