

URNBIS
Internet-Draft
Obsoletes: 2141 (if approved)
Intended status: Standards Track
Expires: January 13, 2014

P. Saint-Andre, Ed.
Cisco Systems, Inc.
R. Moats
July 12, 2013

Uniform Resource Name (URN) Syntax
draft-ietf-urnbis-rfc2141bis-urn-05

Abstract

A Uniform Resource Name (URN) is a Uniform Resource Identifier (URI) that is intended to serve as a persistent, location-independent resource identifier. This document defines the canonical syntax for URIs under the "urn" scheme, guidelines for URN namespaces, requirements for URN presentation and transmission, and methods for determining URN equivalence. This document obsoletes RFC 2141.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 13, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

Table of Contents

1. Introduction	3
2. Terminology	3
3. Requirements	3
4. URN Syntax	3
4.1. Namespace Identifier Syntax	4
4.2. Namespace Specific String Syntax	4
4.3. Query Component and Fragment Identifier Component	5
5. URN Presentation and Transport	5
6. Lexical Equivalence in URNs	6
6.1. Procedure	6
6.2. Examples	6
7. Functional Equivalence in URNs	7
8. Handling of URNs by URI Processors	7
9. Security Considerations	7
10. IANA Considerations	7
11. References	8
11.1. Normative References	8
11.2. Informative References	8
Appendix A. Changes from RFC 2141	9
Appendix B. Acknowledgements	9
Authors' Addresses	9

1. Introduction

A Uniform Resource Name (URN) is a Uniform Resource Identifier (URI) [RFC3986] that is intended to serve as a persistent, location-independent resource identifier. This document defines the canonical syntax for URIs under the "urn" scheme, guidelines for URN namespaces, requirements for URN presentation and transmission, and methods for determining URN equivalence.

URNs were originally defined in [RFC2141]. The goal of this document is to specify URNs with the smallest reasonable set of changes from the original definition while ensuring consistency with the updated specification of URIs in [RFC3986].

This document obsoletes RFC 2141.

2. Terminology

Several important terms used in this document are defined in the URI specification [RFC3986].

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. Requirements

The requirements for URNs are specified in [RFC1737]. This document does not modify or update those requirements.

4. URN Syntax

The syntax for a URN is defined as follows using the Augmented Backus-Naur Form (ABNF) as specified in [RFC5234].

```
namestring    = assigned-name [ "?" query ] [ "#" fragment ]
                ;
                ; query and fragment are defined in RFC 3986
                ;
assigned-name = "urn" ":" NID ":" NSS
                ;
                ; the URI scheme ("urn") is case-insensitive
                ;
NID           = (alphanum) 0*30(ldh) (alphanum)
                ;
                ; alphanum is defined in RFC 3986
                ;
ldh           = alphanum / "-"
NSS          = 1*(pchar)
                ;
                ; pchar is defined in RFC 3986
                ;
```

The following sections describe provide additional information about these rules.

4.1. Namespace Identifier Syntax

The syntax here is slightly more restrictive than what was defined in [RFC2141], since it forbids the character "-" at the end of a NID.

NIDs are case insensitive (e.g., "ISBN" and "isbn" are equivalent).

4.2. Namespace Specific String Syntax

Depending on the rules governing a namespace, names that are valid in a namespace might contain characters that are not allowed in URNs according to the "pchar" rule (e.g., characters outside the ASCII range or characters that are reserved in URIs, such as "/", "?", and "#"). Such a string **MUST** be translated into a conformant NSS before using it as a protocol element or otherwise passing it on to other applications. Translation is done by percent-encoding each disallowed character using the method defined in Section 2.1 of [RFC3986]. Note that the "%" character is allowed only for the purpose of percent-encoding.

If a namespace designates one or more characters conforming to the "pchar" rule as having special meaning for that namespace (e.g., "@") and the namespace also uses that character in a literal sense, when used in a literal sense the character **MUST** be percent-encoded (e.g., "%40"). For related considerations with regard to NID registration, see [I-D.ietf-urnbis-rfc3406bis-urn-ns-reg].

4.3. Query Component and Fragment Identifier Component

The URI specification [RFC3986] allows a query component, a fragment identifier component, or both after the path component of a URI, where the character '?' is used as a separator to denote the beginning of the query component and the character '#' is used as a separator to denote the beginning of the fragment identifier component. The original URN syntax specification [RFC2141] reserved the '?' and '#' characters for future developments. This specification aligns URN syntax with URI syntax by allowing the query component and fragment identifier component after (not within) the Namespace Specific String (NSS).

This specification does not define the applicability and semantics of the query component or the fragment identifier component in URNs. Additional specifications might establish these matters for URN-related services (such as resolution) or for individual URN namespaces. For example, it is possible that the query component might be used in requests to URN resolution services, or that the fragment identifier component might be used to distinguish the integral parts of resources named by URNs. However, defining such usage is left to specifications for URN resolution services, namespace registration requests and specifications for individual namespaces (which might use some namespace-specific syntax instead of the URI fragment identifier component), and other appropriate documentation (such as policy documents governing the management of a given URN namespace).

Although URN assignment is often a managed process (see [I-D.ietf-urnbis-rfc3406bis-urn-ns-reg]), the query component or fragment identifier component can be appended after the NSS once a URN has been assigned in accordance with the rules for a given namespace.

5. URN Presentation and Transport

The URN syntax defines the canonical format for URNs. All URN transport and interchanges MUST take place in this format. Further, all URN-aware applications MUST offer the option of displaying URNs in this canonical form to allow for direct transcription (for example by cut and paste techniques). Such applications might support display of URNs in a more human-friendly form and might use a character set that includes characters that are not permitted in URN syntax as defined in this RFC (i.e., when displaying URNs to humans, such applications might replace percent-encoded strings with characters in an extended character set such as Unicode).

6. Lexical Equivalence in URNs

6.1. Procedure

For various purposes such as caching, often it is desirable to determine if two URNs are "the same". This is done by testing for "lexical equivalence".

Two URNs are lexically equivalent if they are octet-by-octet equal after applying case normalization (as specified in Section 6.2.2.1 of [RFC3986]) to the following constructs:

1. the URI scheme "urn"
2. the NID
3. any percent-encoded characters (see Section 2.1 of [RFC3986])

Percent-encoded characters MUST NOT be decoded, i.e., percent-encoding normalization (as specified in Section 6.2.2.2 of [RFC3986]) MUST NOT be applied.

If a query component, fragment identifier component, or both have been appended to the assigned URI, they MUST be ignored for purposes of determining lexical equivalence.

URN namespaces MAY define additional rules for lexical equivalence, such as case-insensitivity of the NSS (or parts thereof). Such rules MUST always have the effect of eliminating some of the false negatives obtained by the procedure above and MUST NOT result in treating two URNs as not equivalent if the procedure here says they are equivalent. For related considerations with regard to NID registration, see [I-D.ietf-urnbis-rfc3406bis-urn-ns-reg].

6.2. Examples

The following URN comparisons (which use the "example" NID defined in [RFC6963]) highlight the lexical equivalence rules:

1. URN:example:a123,456
2. urn:example:a123,456
3. urn:EXAMPLE:a123,456
4. urn:example:A123,456
5. urn:example:a123%2C456
6. URN:EXAMPLE:a123%2c456

URNs 1, 2, and 3 are lexically equivalent. URN 4 is not lexically equivalent to any of the other URNs in the above set. URNs 5 and 6 are lexically equivalent only to each other.

7. Functional Equivalence in URNs

Functional equivalence is determined within a given namespace and managed by resolvers for that namespace, and thus is beyond the scope of this document. For related considerations with regard to NID registration, see [I-D.ietf-urnbis-rfc3406bis-urn-ns-reg].

8. Handling of URNs by URI Processors

The URN syntax has been defined so that URNs can be used in places where URIs are expected. A resolver that conforms to the URI specification [RFC3986] will extract a scheme of "urn" rather than a scheme value of "urn:<nid>".

A URN MUST be considered an opaque URI by URI resolvers and passed (with the "urn" scheme) to a URN resolver for resolution. The URN resolver can either be an external resolver that the URI resolver knows of, or it can be functionality built-in to the URI resolver.

To minimize user confusion, a URI browser SHOULD display the complete URN (including the "urn" scheme) to ensure that there is no confusion between URN namespace identifiers and URL scheme identifiers.

9. Security Considerations

This document specifies the syntax for URNs. While some namespaces resolvers might assign special meaning to certain of the characters of the Namespace Specific String, any security considerations resulting from such assignment are outside the scope of this document. For related considerations with regard to NID registration, see [I-D.ietf-urnbis-rfc3406bis-urn-ns-reg].

10. IANA Considerations

This section formally registers a URI scheme of 'urn'.

[Note to RFC Editor: please replace "XXXX" with the number assigned to this document upon publication.]

URI Scheme Name: urn
Status: permanent

URI Scheme Syntax: See Section 4 of RFCXXXX.

URI Scheme Semantics: The 'urn' scheme identifies Uniform Resource Names, which are persistent, location-independent resource identifiers.

Encoding Considerations: See Section 4.2 of RFCXXXX.

Applications/Protocols That Use This URI Scheme Name: Uniform Resource Names are used in a wide variety of applications, including bibliographic reference systems and as names for Extensible Markup Language (XML) namespaces.

Interoperability Considerations: There are no known interoperability concerns related to use of the 'urn' URI scheme.

Security Considerations: See Section 9 of RFCXXXX.

Contact: URNBIS WG [mailto:urn@ietf.org]

Author/Change Controller: This scheme is registered under the IETF tree. As such, the IETF maintains change control.

References None.

11. References

11.1. Normative References

- [I-D.ietf-urnbis-rfc3406bis-urn-ns-reg]
Saint-Andre, P., Daigle, L., Iannella, R., and P. Faltstrom, "Uniform Resource Name (URN) Namespace Definition Mechanisms", draft-ietf-urnbis-rfc3406bis-urn-ns-reg-06 (work in progress), July 2013.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, January 2005.
- [RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.

11.2. Informative References

- [RFC1737] Sollins, K. and L. Masinter, "Functional Requirements for Uniform Resource Names", RFC 1737, December 1994.
- [RFC2141] Moats, R., "URN Syntax", RFC 2141, May 1997.
- [RFC6963] Saint-Andre, P., "A Uniform Resource Name (URN) Namespace for Examples", BCP 183, RFC 6963, May 2013.

Appendix A. Changes from RFC 2141

This document makes the following substantive changes from [RFC2141]:

- o Allowed the URI query component after the URN as assigned.
- o Allowed the URI fragment identifier component after the URN as assigned.
- o Disallowed "-" at the end of a NID.
- o Allowed the "~" and "&" characters in an NSS.
- o Formally registered 'urn' as a URI scheme.

Appendix B. Acknowledgements

RFC 2141, which provided the basis for this document, was authored by Ryan Moats.

Many thanks to Leslie Daigle, Martin Duerst, Juha Hakala, Ted Hardie, Alfred Hoenes, John Klensin, Keith Moore, Julian Reschke, Lars Svensson, and other participants in the URNBIS WG for their input.

Julian Reschke provided helpful corrections to the ABNF.

Authors' Addresses

Peter Saint-Andre (editor)
Cisco Systems, Inc.
1899 Wynkoop Street, Suite 600
Denver, CO 80202
USA

Phone: +1-303-308-3282
Email: psaintan@cisco.com

Ryan Moats

URNBIS
Internet-Draft
Obsoletes: 3406 (if approved)
Intended status: BCP
Expires: January 13, 2014

P. Saint-Andre, Ed.
Cisco Systems, Inc.
L. Daigle
Thinking Cat Enterprises
D. van Gulik
WebWeaving
R. Iannella
Semantic Identity
P. Faltstrom
Netnod
July 12, 2013

Uniform Resource Name (URN) Namespace Definition Mechanisms
draft-ietf-urnbis-rfc3406bis-urn-ns-reg-06

Abstract

This document supplements the Uniform Resource Name (URN) syntax specification by defining the concept of a URN namespace, as well as mechanisms for defining and registering such namespaces. This document obsoletes RFC 3406.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 13, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Terminology	3
3. What is a URN Namespace?	4
4. URN Namespace Types	5
4.1. Formal Namespaces	5
4.2. Informal Namespaces	6
5. Defining a URN Namespace	6
5.1. Formal Namespaces	7
5.1.1. Syntax	7
5.1.2. Specification	7
5.2. Informal Namespaces	8
6. Registering a URN Namespace	9
6.1. Formal Namespaces	9
6.2. Informal Namespaces	9
7. URN Namespace Definition Template	10
8. Security Considerations	15
9. IANA Considerations	15
10. References	15
10.1. Normative References	15
10.2. Informative References	16
Appendix A. Changes from RFC 3406	16
Authors' Addresses	16

1. Introduction

A Uniform Resource Name (URN) [I-D.ietf-urnbis-rfc2141bis-urn] is a Uniform Resource Identifier (URI) [RFC3986] that is intended to serve as a persistent, location-independent resource identifier. This document supplements the Uniform Resource Name (URN) syntax specification [I-D.ietf-urnbis-rfc2141bis-urn] by defining the following:

- o The concept of a URN namespace.
- o A mechanism for defining URN namespaces and associating each namespace with a public identifier (called a Namespace ID or "NID").
- o Procedures for registering namespace identifiers with the Internet Assigned Numbers Authority (IANA).

This document rests on two key assumptions:

1. Assignment of a URN is a managed process.

A string that conforms to the URN syntax is not necessarily a valid URN, because a URN needs to be assigned according to the rules of a particular namespace (in terms of syntax, semantics, and process).

2. The space of URN namespaces is itself managed.

A string in the namespace identifier slot of the URN syntax is not necessarily a valid URN namespace identifier, because in order to be valid a namespace needs to be defined and registered in accordance with the rules of this document.

URN namespaces were originally defined in [RFC2611], which was obsoleted by [RFC3406]. Based on experience with defining and registering URN namespaces since that time, this document specifies URN namespaces with the smallest reasonable set of changes from [RFC3406]. This document obsoletes RFC 3406.

2. Terminology

Several important terms used in this document are defined in the URN syntax specification [I-D.ietf-urnbis-rfc2141bis-urn].

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. What is a URN Namespace?

For the purposes of URNs, a "namespace" is a collection of unique identifiers that are consistently assigned according to a common definition.

The uniqueness constraint means that an identifier within the namespace is never assigned to more than one resource and never re-assigned to a different resource (however, a single resource can have more than one URN assigned to it for different purposes).

The consistent assignment constraint means that an identifier within the namespace is assigned by an organization or in accordance with a process that is always followed (e.g., in the form of an algorithm).

The common definition constraint means that both the syntax for identifiers within the namespace and the process for assigning such identifiers are clearly defined in a specification.

A URN namespace is identified by a particular designator (which syntactically follows the 'urn' scheme name) in order to:

- o Ensure the global uniqueness of URNs.
- o Optionally provide a cue regarding the structure of URNs assigned within a namespace.

With regard to global uniqueness, using different designators for different collections of identifiers ensures that no two URNs will be the same for different resources (since each collection is required to uniquely assign each identifier). For instance, some identifier systems use strings of numbers as identifiers (e.g., ISBN, ISSN, phone numbers). It is conceivable that some numbers might be valid identifiers in two different established identifier systems, where the namespace identifier differentiates between the resulting URNs.

With regard to the structure of URNs assigned within a namespace, the development of an identifier structure, and thereby a collection of identifiers, is a process that is inherently dependent on the requirements of the community defining the identifiers, how they will be assigned, and the uses to which they will be put. All of these issues are specific to the individual community seeking to define a namespace (e.g., a publishing community, an association of booksellers, developers of particular application protocols, etc.); therefore these issues are beyond the scope of URN syntax and the rules regarding URN namespaces in general.

URN namespaces inherit certain rights and responsibilities, including:

- o They uphold the general principles of a well-managed URN namespace by providing persistent identification of resources and unique assignment of identifier strings.
- o They can be registered in global registration services.

4. URN Namespace Types

There are two types of URN namespace: formal and informal. These are distinguished by the expected level of service, the information necessary to define the namespace, and the procedures for registration. To date, the vast majority of the registered namespaces have been formal, so this document concentrates on formal namespaces.

Note: [RFC3406] defined a third type of "experimental namespaces", denoted by prefixing the namespace identifier with the string "X-". Consistent with [RFC6648], this specification removes the experimental category.

4.1. Formal Namespaces

A formal namespace can be requested, and IETF review sought, in cases where the publication of the NID proposal and the underlying namespace will provide benefit to some subset of users on the Internet. That is, a formal NID proposal, if accepted, needs to be functional on and with the global Internet, not limited to users in communities or networks not connected to the Internet. For example, consider a NID that is meant for naming of physics research; if that NID request effectively forced someone to use a proprietary network or service that was not at all open to the general Internet user, then it would make a poor request for a formal NID. The intent is that, while the community of those who might actively use the names assigned within that NID might be small (but no less important), the potential use of names within that NID is open to any user on the Internet.

It is expected that formal NIDs might be applied to namespaces where some aspects are not fully open. For example, a namespace might make use of a fee-based, privately managed, or proprietary registry for assignment of URNs in the namespace. However, it might still provide benefit to some Internet users if the services associated have openly-published access protocols.

In addition to the basic information specified in the namespace definition template (see Section 7), a formal namespace request needs to be accompanied by documented considerations of the need for a new namespace and of the community benefit from formally establishing the

proposed URN namespace.

Additionally, since the goal of URNs is to provide persistent identification, a formal namespace request needs to give some consideration as to the longevity and maintainability of the namespace. Possible factors to consider with regard to an organization that will assign URNs within a namespace include the following:

- o It ought to demonstrate stability and the ability to maintain the URN namespace for a long time; absent such evidence, it ought to be clear how the namespace can remain viable if the organization can no longer maintain the namespace.
- o It ought to demonstrate competency in name assignment. This will improve the likelihood of persistence (e.g. to minimize the likelihood of conflicts).
- o It ought to commit to not re-assigning existing names and to allowing old names to continue to be valid, even if the owners or assignees of those names are no longer members or customers of that organization. With regard to URN resolution, this does not mean that there needs to be resolution of such names, only that the names will not resolve to false or stale information.

4.2. Informal Namespaces

Informal namespaces are full-fledged URN namespaces, with all the rights and responsibilities associated thereto. Informal namespaces differ from formal namespaces in the process for assigning a NID: IANA will assign an alphanumeric NID (e.g., "urn-7") to informal namespaces, per the process outlined under Section 6.

5. Defining a URN Namespace

A URN namespace is defined by the following factors:

- o The syntax of URNs assigned within the namespace, in conformance with the fundamental URN syntax [I-D.ietf-urnbis-rfc2141bis-urn].
- o The process for assigning URNs within the namespace.
- o Optionally, the process for resolving URNs issued within the namespace.

Processes for resolution of URNs assigned within a namespace (if any) are out of scope for this document. The following sections provide guidelines for (1) defining the syntax of URNs within a namespace and (2) specifying how URNs will be assigned within a namespace.

5.1. Formal Namespaces

Formal NIDs are assigned as a result of IETF Review as defined in the "IANA Considerations" document [RFC5226]. Thus an application for a formal NID is made by publishing an RFC in the IETF stream, either as the product of an IETF working group or as an individual submission sponsored by an Area Director. The RFC need not be standards track (indeed, to date most RFCs registering URN namespaces have been informational), but it will be subject to IESG review and approval pursuant to the guidelines provided here (as well as standard RFC publication guidelines).

5.1.1. Syntax

A formal namespace registration requests a particular NID, subject to the following constraints (above and beyond the syntax rules specified in [I-D.ietf-urnbis-rfc2141bis-urn]):

- o It MUST NOT be an already-registered NID.
- o It MUST NOT start with "urn-" (which is reserved for informal namespaces).
- o It MUST be more than two characters long.
- o It MUST NOT start with "XY-", where "XY" is any combination of two ASCII letters.

All two-letter combinations, and all two-letter combinations followed by "-" and any sequence of valid NID characters, are reserved for potential use as countrycode-based NIDs for eventual national registrations of URN namespaces. The definition and scoping of rules for allocation of responsibility for such countrycode-based namespaces is beyond the scope of this document.

5.1.2. Specification

The specification defining a formal namespace MUST include a completed namespace definition template (see Section 7).

The specification also MUST include the following sections.

First, the "Namespace Considerations" section outlines the perceived need for a new namespace (e.g., by describing where existing namespaces fall short of the proposer's requirements). Potential considerations include:

- o The type of resources to be identified
- o The type of services to be supported

- o Procedures for assigning URNs within this namespace
- o Processes for resolving URNs assigned within this namespace, if any

It is expected that more than one namespace might serve the same "functional" purpose; the intent of the "Namespace Considerations" section is to provide a record of the proposer's "due diligence" in exploring existing possibilities, for the consideration by the Internet community, expert reviewers, and the IESG.

Second, the "Community Considerations" section explains how the intended community will benefit by assignment of this namespace, as well as how a general Internet user will be able to use the space if they care to do so. Potential considerations include:

- o Methods and benefits for using the assigned URNs
- o Methods and benefits for resolving the assigned URNs (if any)
- o The kinds of software applications that can use or resolve the assigned URNs (e.g., by differentiating among disparate namespaces, identifying resources in a persistent fashion, or meaningfully resolving and accessing services associated with the namespace)

Third, the "Security Considerations" section describes any potential security-related issues with regard to assignment, use, and resolution of identifiers within the namespace. Examples of such issues include the consequences of producing false negatives and false positives during comparison for lexical equivalence (see also [RFC6943]), leakage of private information when identifiers are communicated on the public Internet, the potential for directory harvesting, and the issues discussed in [RFC3552].

Fourth, the "IANA Considerations" section indicates that the document includes a URN NID registration that is to be entered into the IANA registry of URN NIDs.

5.2. Informal Namespaces

Informal namespaces are directly requested of IANA and are assigned based on a policy of First Come First Served [RFC5226].

The namespace identifier assigned by IANA has the following syntax:

"urn-" <number>

The <number> is chosen by IANA. The only restrictions on <number> are that it (1) consist strictly of ASCII digits and (2) not cause the NID to exceed the length limitations defined in the URN syntax

specification [I-D.ietf-urnbis-rfc2141bis-urn].

6. Registering a URN Namespace

6.1. Formal Namespaces

The registration policy for formal namespaces is IETF Review [RFC5226]. The key steps for registration of a formal namespace are:

1. Submit an Internet-Draft that includes all of the information described under Section 5.1.2 and Section 7 of this document.
2. Send the completed namespace definition template, along with a pointer to the Internet-Draft, to the urn-nid@ietf.org discussion list for technical review.
3. If necessary to address comments received, repeat steps 1 and 2.
4. Ask the responsible Area Director to process the Internet-Draft for publication as an RFC. Note that the IESG can request further changes or direct discussion to designated working groups, area experts, etc.
5. If the IESG approves the document for publication as an RFC, the IANA will register the requested NID.

A registration can be revised by updating the RFC through normal IETF processes [RFC2606]. The authors of the revised document need to follow the same steps outlined above for new registrations.

6.2. Informal Namespaces

The registration policy for informal namespaces is First Come First Served [RFC5226]. The key steps for registration of an informal namespace are:

1. Write a completed namespace definition template (see Section 7). This can be done as part of an Internet-Draft.
2. Send the completed template to the urn-nid@ietf.org discussion list for technical review.
3. If necessary to address comments received, repeat steps 1 and 2.
4. Once comments have been addressed and the review period has expired, send a registration request to IANA (via the iana@iana.org email address) with the final template.

Informal namespaces can also be revised by updating the template and processing it as outlined above for new registrations.

7. URN Namespace Definition Template

Definition of a URN namespace is accomplished by completing the following template. In addition to providing a mechanism for defining the structure of URNs assigned within the namespace, this information is designed to be useful for:

- o entities seeking to have a URN assigned in a namespace (if applicable)
- o entities seeking to provide URN resolvers for a namespace (if applicable)

Providing a complete and accurate template is particularly helpful to communities that are evaluating the possibility of using a portion of an existing URN namespace rather than creating a new namespace.

As described under Section 5.1.2, applications for formal URN namespaces MUST also document the "Namespace Considerations", "Community Considerations", "Security Considerations", and "IANA Considerations".

The information to be provided in the template is as follows:

Namespace ID:

Requested of IANA (formal) or assigned by IANA (informal).

Registration Information:

The version and date of the registration:

- Registration version number: starting with 1, incrementing by 1 with each new version
- Registration date: date submitted to the IANA, using the format YYYY-MM-DD

Declared registrant of the namespace:

This includes:

- Registering organization
 - Name
 - Address
- Designated contact person
 - Name
 - Contact information
 - (at least one of email address,

phone number, postal address)

Declaration of syntactic structure:

This section ought to outline any structural features of identifiers in this namespace. At the very least, this description can be used to introduce terminology used in other sections. This structure can also be used for determining realistic caching/shortcuts approaches; suitable caveats ought to be provided. If there are any specific character encoding rules (e.g., which character ought to always be used for single-quotes), these ought to be listed here. If the namespace allows use of the URI query component, URI fragment identifier component, or both, such usage needs to be described here (in addition to any other namespace-specific syntax, such as distinguishers for integral parts of resources identified by URNs within the namespace).

At a high level, answers might include, but are not limited to:

- A formal definition of the structure, e.g., in terms of Augmented BNF for Syntax Specifications (ABNF) as specified in [RFC5234]
- A regular expression for parsing the identifier into components, including naming authorities
- An algorithm for generating conformant URNs
- An explanation that the structure is opaque

Relevant ancillary documentation:

This section ought to list any RFCs, specifications, or other published documentation that defines or explains all or part of the namespace structure.

At a high level, answers might include, but are not limited to:

- Pointers to specifications that define the syntax and semantics of the namespace
- Mention of documentation that describes the processes followed by an organization that assigns URNs in the namespace
- Explanatory material describing the namespace

Identifier uniqueness considerations:

This section ought to address the requirement that URNs are assigned uniquely -- i.e., they are assigned to at most one

resource, and are not reassigned.

(Note that the definition of "resource" is fairly broad; for example, information on "Today's Weather" might be considered a single resource, although the content is dynamic.)

At a high level, answers might include, but are not limited to:

- Exposition of the structure of the identifiers, and partitioning of the space of identifiers amongst assignment authorities which are individually responsible for respecting uniqueness rules
- Description of a method for assignment of identifiers (e.g., identifiers are assigned sequentially)
- An explanation that this information is withheld (i.e., the namespace is opaque)

Identifier persistence considerations:

Although non-reassignment of URN identifiers ensures that a URN will persist in identifying a particular resource even after the "lifetime of the resource", some consideration ought to be given to the persistence of the usability of the URN. This is particularly important in the case of URN namespaces providing global resolution.

At a high level, answers could include, but are not limited to:

- Quality of service considerations

Process of identifier assignment:

This section ought to detail the mechanisms and/or authorities for assigning URNs to resources. It ought to make clear whether assignment is completely open or, if limited, how to become an assigner of identifiers or how to get an identifier assigned by existing assignment authorities.

At a high level, answers could include, but are not limited to:

- Assignment is completely open, following a particular algorithm
- Assignment is delegated to authorities recognized by a particular organization (e.g., the Digital Object Identifier Foundation controls the DOI assignment space and its delegation)
- Assignment is completely closed (e.g., for a private organization)

Process for identifier resolution:

If a namespace is intended to be accessible for global resolution, it needs to be registered in an RDS (Resolution Discovery System, see [RFC 2276]) such as DDDS. Resolution then proceeds according to standard URI resolution processes, and the mechanisms of the RDS. What this section ought to outline is the requirements for becoming a recognized resolver of URNs in this namespace (and being so listed in the RDS registry).

At a high level, answers might include, but are not limited to:

- The namespace is not listed with an RDS; therefore this section is not applicable
- Resolution mirroring is completely open, with a mechanism for updating an appropriate RDS
- Resolution is controlled by entities to which assignment has been delegated

Rules for lexical equivalence:

If there are particular algorithms for determining equivalence between two identifiers in the underlying namespace (hence, in the URN string itself), rules can be provided here. Such rules ought to always have the effect of eliminating false negatives that might otherwise result from comparison.

If it is appropriate and helpful to do so, reference can be made to the equivalence rules defined in the URI specification [RFC3986].

Some examples include:

- Equivalence between uppercase and lowercase characters in the Namespace Specific String
- Equivalence between hyphenated and non-hyphenated groupings in the identifier string
- Equivalence between single-quotes and double-quotes
- Namespace-defined equivalences between specific characters, such as "character X with or without diacritic marks".

Note that these are not normative statements for any kind of best practice related to handling of equivalences between characters in general; they are statements limited in scope to reflecting the rules for this specific namespace only.

Conformance with URN syntax:

This section ought to outline any special considerations necessary for conforming with the URN syntax. This is particularly applicable in the case of legacy naming systems that are used in the context of URNs.

For example, if a namespace is used in contexts other than URNs, it might make use of characters that are reserved in the URN syntax.

This section ought to flag any such characters, and outline necessary mappings to conform to URN syntax. Normally, this will be handled by percent-encoding the character as specified in the URI specification [RFC3986].

Validation mechanism:

Apart from attempting resolution of a URN, a URN namespace may provide mechanisms for "validating" a URN -- i.e., determining whether a given string is currently a validly-assigned URN. There are two issues here: 1) users ought not "guess" URNs in a namespace; 2) when the URN namespace is based on an existing identifier system, it might not be the case that all existing identifiers are assigned on Day 0. The reasonable expectation is that the resource associated with each resulting URN is somehow related to the thing identified by the original identifier system, but those resources might not exist for each original identifier. For example, even if a URN namespace were defined based on telephone numbers, it is not clear that all telephone numbers would immediately become "valid" URNs resolvable using whatever mechanisms are described as part of the namespace registration.

Validation mechanisms might be:

- A syntax grammar
- An online service
- An offline service

Scope:

This section ought to outline the scope of the use of the identifiers in this namespace. Apart from considerations of private vs. public namespaces, this section is critical in evaluating the applicability of a requested NID. For example, a namespace claiming to deal in "social security numbers" ought to have a global scope and address all social security number structures (unlikely). On the other hand, at a national level, it is reasonable to propose a URN namespace for "this

nation's social security numbers".

8. Security Considerations

This document largely focuses on providing mechanisms for the declaration of public information. Nominally, these declarations will be of relatively low security profile, however there is always the danger of "spoofing" and providing misinformation. Information in these declarations ought to be taken as advisory.

The definition of a URN namespace needs to account for potential security issues related to assignment, use, and resolution of identifiers within the namespace; see Section 5.1.2 for further discussion.

9. IANA Considerations

This document outlines the processes for registering URN namespaces, and has implications for the IANA in terms of registries to be maintained. In all cases, the IANA ought to assign the appropriate NID (formal or informal) once the procedures outlined in this document have been completed.

10. References

10.1. Normative References

- [I-D.ietf-urnbis-rfc2141bis-urn]
Saint-Andre, P. and R. Moats, "Uniform Resource Name (URN) Syntax", draft-ietf-urnbis-rfc2141bis-urn-05 (work in progress), July 2013.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, January 2005.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

10.2. Informative References

- [RFC2606] Eastlake, D. and A. Panitz, "Reserved Top Level DNS Names", BCP 32, RFC 2606, June 1999.
- [RFC2276] Sollins, K., "Architectural Principles of Uniform Resource Name Resolution", RFC 2276, January 1998.
- [RFC2611] Daigle, L., van Gulik, D., Iannella, R., and P. Faltstrom, "URN Namespace Definition Mechanisms", BCP 33, RFC 2611, June 1999.
- [RFC3406] Daigle, L., van Gulik, D., Iannella, R., and P. Faltstrom, "Uniform Resource Names (URN) Namespace Definition Mechanisms", BCP 66, RFC 3406, October 2002.
- [RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, July 2003.
- [RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.
- [RFC6648] Saint-Andre, P., Crocker, D., and M. Nottingham, "Deprecating the "X-" Prefix and Similar Constructs in Application Protocols", BCP 178, RFC 6648, June 2012.
- [RFC6943] Thaler, D., "Issues in Identifier Comparison for Security Purposes", RFC 6943, May 2013.

Appendix A. Changes from RFC 3406

Although on the surface it might appear that this document is significantly different from [RFC3406], in general it only modifies the order of presentation, with the intent of making it easier for interested parties to define and register URN namespaces. In addition, some of the text was updated to be consistent with the definition of Uniform Resource Identifiers (URIs) [RFC3986] and the processes for registering information with the IANA [RFC5226], as well as more modern guidance with regard to security issues [RFC3552] and identifier comparison [RFC6943]. The only major substantive change was removing the category of experimental namespaces, consistent with [RFC6648].

Authors' Addresses

Peter Saint-Andre (editor)
Cisco Systems, Inc.
1899 Wynkoop Street, Suite 600
Denver, CO 80202
USA

Phone: +1-303-308-3282
Email: psaintan@cisco.com

Leslie Daigle
Thinking Cat Enterprises

Dirk-Willem van Gulik
WebWeaving

Renato Iannella
Semantic Identity

Patrick Faltstrom
Netnod

