Names are Not Locators and URNs are Not URIs
draft-ietf-urnbis-urns-are-not-uris-01.txt

Abstract

   Experience has shown that identifiers associated with persistent
   names are quite different from identifiers associated with the
   locations of objects. This is especially true when such names are
   are expected to be stable for a very long time or when they identify
   large and complex entities. In order to allow Uniform Resource Names
   (URNs) to evolve to meet the needs of the Informational Sciences
   community and other users, this specification separates the syntax
   for URNs from the generic syntax for Uniform Resource Identifiers
   (URIs) specified in RFC 3986, updating the latter specification
   accordingly.

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 03, 2015.

Copyright Notice

Table of Contents

1. Introduction

   The Internet community now has many years of experience with both
   name-type identifiers and location-based identifiers (or "references"
   for those who are sensitive to the term "identifier" -- see Section
   3). The primary examples of these two categories are Uniform
   Resource Names (URNs [RFC2141] [RFC2141bis]) and Uniform Resource
   Locators (URLs) [RFC1738]). That experience leads to the conclusion
   that it is impractical to constrain URNs to the syntax and high-level
   semantics of URLs. Generalization from URLs to generic Uniform
   Resource Identifiers (URIs) [RFC3986], especially to name-based,
   high-stability, long-persistence, identifiers such as many URNs, has
   failed because the assumed similarities do not actually exist to a
   sufficient degree. Ultimately, locators, which typically depend on
   particular accessing protocols and a specification relative to some
   physical space or network topology, are simply different creatures
   from long-persistence, location-independent, object identifiers. The
   syntax and semantic constraints that are appropriate for locators are
   either irrelevant to or interfere with the needs of resource names as
   a class. That was tolerable as long as the URN system didn't need
   additional capabilities but experience since RFC 2141 was published
   has shown that they are, in fact, needed.

   Even then, it would have been possible to make URNs fit the "generic
   URI" [RFC3986] bed by inventing a syntax with sufficient escapes and
   embedding had the latter specification not also specified some
   semantics for non-locational information. Whether such escapes and
   embedding would have been a good idea is another matter: they tend to
   make syntax more complex, harder for users to understand, and hence

   more error-prone.

   This specification updates the Generic URI Syntax specification
   [RFC3986] to exclude URNs from its coverage. Put differently, with
   the publication of this specification, URNs are no longer considered
   a member of the class of URIs to which RFC 3986 applies.

   [[Note in draft: the above leaves it ambiguous as to whether it
   remains appropriate to call URNs "URIs". That ambiguity is
   intentional and, if possible should keep the question part of the
   "someone else's problem" category.]]

   For URLs and such other URIs as may exist or be created in the
   future, this specification does not change the syntax rules and other
   requirements and recommendations of RFC 3986.

2. Pragmatic Goals

   Despite the important background and rationale in the section that
   follows, the change made by this specification is driven by a desire
   to avoid philosophical debates about terminology or ultimate truths.
   Instead, it is motivated by three very pragmatic principles:

   1. Try to accommodate all of those who think URNs are necessary,
       i.e., that they are distinct from URLs.

   2. Try to avoid getting bogged down in declarative/ definitional
       statements about what is and is not correct in the abstract.

   3. Avoid a fork in the standard that leads to multiple, conflicting,
       definitions or criteria for URNs.

3. A Perspective on Locations and Names

   [[Note in Draft: See Appendix A for a different perspective.]]

   Content industries (e.g., publishers) and memory organizations (e.g.,
   libraries, archives, and museums) invest a lot of resources on naming
   things and the topics of naming and classification are important
   information science issues. Tens, if not hundreds, of millions of
   persistent identifiers have been assigned during the last decade.

   Several identifier systems have been developed for persistent and
   unique identification of resources. When there is a real need to
   preserve something important (such as scientific publications,
   research data, government publications, etc.) for the long term, URNs
   or other persistent identifiers are used; URLs (or other generic
   URIs) are not being used for identification or even linking purposes.

Naming and locating, e.g., for library resources, are both complex
activities which have different aims. Traditionally, naming and
locating resources have been separate activities, and the rules for
the former are much more stringent than for the latter. The same
principles are being applied to digital materials as well as more
traditional ones. In a library, any book, be it printed or digital,
has both unique and persistent International Standard Book Number
(ISBN) and non-unique (each copy has its own location information)
and short-lived location information which cannot be trusted in the
long run. ISBN never changes, but both shelf locations and Web
addresses usually do, many times during the book's life span.

Giving location information a role in identification would not only
force libraries to adopt different policies for printed and digital
content, it would also undermine the value of existing identifier
systems. Let us assume that ten people independently upload a copy
of an electronic book into different locations in the Web. Are all
these ten URLs valid identifiers of the book? And what is their
relation to the ISBN or other identification information of the book
such as its title?

From the perspective of the communities who depend on persistent
identifiers, critical issues include:

1. Resource identification has to be a managed process. Assigning
    URIs generally is not. Although it may be possible to introduce
    some level of control to URI assignment, a user cannot determine
    whether some URI is reliable or not.

2. Anyone may assign new URIs to resources even if these resources
    already have proper identifiers assigned to them. Claiming that
    these URIs actually identify something undermines the value of
    proper identifiers.

3. There is no 1:1 relation between the resource identified and
    URIs. An e-book in the Web may be represented as 1-n files
    (URIs), and a single file may contain several books. And books
    are simple, we need to name very complex objects such as research
    data sets, or some component parts within these complex data
    sets.

4. One resource such as a scientific article is typically available
    from multiple locations, including (for instance) the publisher's
    document supply service, a university's open repositories and
    other cooperative repository systems, legal deposit collections
    and the Internet archive. A resource should have one and only
    one identifier of a given type; URIs do not meet this
    requirement.

5. URIs relate to instances (copies) of resources, whereas
    traditionally identification has much broader scope. Identifiers
    may be assigned to, e.g., an immaterial work (such as Hamlet),
    its expressions (e.g. Finnish translation of Hamlet), and
    manifestations of works and expressions (e.g. PDF version of
    Finnish translation of Hamlet).

6. Over time, different resources (or different versions of the same
    resource) may be found from the same non-URN URI. A user has no
    way of knowing whether the resource has changed. One of the
    basic principles for proper identifier systems is that the same
    identifier is never assigned to another resource. In general,
    URIs do not meet this requirement.

7. Persistent identification must be available for resources which
    are available only in databases and other environments that are
    often identified today as "deep web". URIs for these resources
    tend to be very complicated and it will be difficult to keep them
    alive even with the help of DNS redirection when e.g. the
    underlying database management system changes.

8. The role URI fragment and query could or should have in
    identification is unclear and the statements in RFC 3986 are
    definitely problematic from the points of view of existing
    identifier systems and management of naming.

       Does "fragment" identify a location or a certain section of a
       resource? In the evolving set of URN Internet standards,
       fragment will not be a part of the Namespace Specific String.
       Then fragment only indicates a place / segment within the
       identified resource, but does not identify it. If fragment
       had a role in identification, fragments would extend the scope
       of existing standard identifiers to component parts of
       resources. For instance, anyone could use URN based on ISBN +
       fragment to identify chapters of electronic books.

       Things get even more complicated with "query" since what the
       combination of an identifier and a query resolves to may not
       have anything to do with the original resource. For instance,
       a URN based in ISBN + query may resolve to the metadata record
       describing the book. These records have their own identifiers
       which are not based on ISBNs.

       [[Note in draft: Most of the discussion above may belong in
       2141bis and/or 3406bis rather than here.]]

9. For many organizations, persistence means decades or centuries.
    Anything that is protocol dependent will eventually fail. URLs
    do not change by themselves, but in the long run it is very
    difficult for people to not change them or the objects to which
    they point.

The mention of centuries is intentional. Content industries, memory organizations (such as national and repository libraries and national archives) and universities and other research organizations, need identifiers that will persist for hundreds of years. Such identifiers might even need to outlast the institutions themselves, and definitely should be usable even if current technologies such as the Web and the Internet cease to exist or are supplanted by something new (as unlikely as that might seem today).

In addition, operations on, or additional specifications about, names and the associated objects must be possible, as stable as the names themselves, and reasonably efficient. For example, if a URN were assigned to an encyclopedia that consisted of many volumes, it should be feasible to identify (and locate and retrieve if that were desired) a particular volume or even a particular article without accessing or retrieving the entire set.

4. Changes to RFC 3986

This specification removes URNs from the scope of RFC 3896. It makes no changes for URI types that remain within that scope and has no practical effect for URNs defined in strict conformance to the prior URN specification [RFC2141] or the associated registration specification [RFC3406].

5. Other Required Actions

The basic URN syntax specification [RFC2141] was published well before RFC 3986 and therefore does not depend on it. Successors to that specification will need to fully spell out the syntax and semantics of URNs, eliminating or using great care about generic or implicit reference to any URI specification.

6. Acknowledgments

This specification was inspired by a search in the IETF URNBIS WG for other alternatives that would both satisfy the needs of persistent name-type identifiers and still fully conform to the specifications and intent of RFC 3986. That search lasted several years and considered many alternatives. Discussions with Leslie Daigle, Juha Hakala, Barry Leiba, Keith Moore, Andrew Newton, and Peter Saint-Andre during the last quarter of 2013 and the first quarter of 2014 were particularly helpful in getting to the conclusion that a conceptual separation of notions of location-based identifiers (e.g., URLs) and the types of persistent identifiers represented by URNs was necessary. As noted below, Juha Hakala provided much of the text on which Section 3 was based. Peter Saint-Andre provided significant text in a pre-publication review. The author also appreciates the efforts of several people, notably Tim Berners-Lee, Julian Reschke, Lars Svensson, Henry S. Thompson, and Dale Worely, to challenge text and ideas and demand answers to hard questions. Whether they agree with the results or not, their insights have contributed significantly to whatever clarity and precision appears in the text.

7. Contributors

   Juha Hakala contributed most of the text of Section 3.

      Contact Information:
   Juha Hakala
   The National Library of Finland
   P.O. Box 15, Helsinki University
   Helsinki, MA FIN-00014
   Finland
   Email: juha.hakala@helsinki.fi

8. IANA Considerations

   [[RFC Editor: Please remove this section before publication.]]

   This memo is not believed to require any action on IANA's part. In particular, we note that there are a collection of "Uniform Resource Identifier (URI) Schemes" that does not include URNs and a series of URN-specific registries that do not rely on the URI specificstions.

9. Security Considerations

   This specification changes the structural syntax and semantics of URNs to make them self-contained (as specified in other documents) rather than making them dependent on generic URI syntax. It should have no effect on Internet security unless the use of a definition and syntax that are more clear reduces the potential for confusion and consequent vulnerabilities.

10. References

10.1. Normative References

   [RFC2141] Moats, R., "URN Syntax", RFC 2141, May 1997.

   [RFC3986] Berners-Lee, T., Fielding, R. and L. Masinter, "Uniform
              Resource Identifier (URI): Generic Syntax", STD 66, RFC
              3986, January 2005.

10.2. Informative References

   [DeterministicURI]
              Mazahir, O., Thaler, D. and G. Montenegro, "Deterministic
              URI Encoding", February 2014, <http://www.ietf.org/id/
              draft-montenegro-httpbis-uri-encoding-00.txt>.

   [RFC1738] Berners-Lee, T., Masinter, L. and M. McCahill, "Uniform
              Resource Locators (URL)", RFC 1738, December 1994.

   [RFC2141bis]
              Saint-Andre, P., "Uniform Resource Name (URN) Syntax",
              January 2014, <https://datatracker.ietf.org/doc/draft-
              ietf-urnbis-rfc2141bis-urn/>.

   [RFC3406] Daigle, L., van Gulik, D., Iannella, R. and P. Faltstrom,
              "Uniform Resource Names (URN) Namespace Definition
              Mechanisms", BCP 66, RFC 3406, October 2002.

Appendix A. A More Pragmatic Perspective

   [[The community should decide whether this appendix, or a modified
   version of it, should remain or be removed at the time of RFC
   publication. In principle, it could even be retained by splitting
   the relevant Section above into two parts and making a variation on
   the text below into one of them.   Those who think it should be
   retained are encouraged to supply text.]]

   Section 3 provides an explanation of the reasons for this change.
   That explanation is not without controversy, especially from those
   who make different assumptions about the future, or even
   interpretations of the present, than many members of the community
   (and especially members of the communities described in that
   section). Some of those who do not accept the explanation above
   simply do not recognize the distinctions on which it, and URNs more
   generally, are based, including the name-locator distinction. In
   some cases, opposition to that explanation is quite pronounced,

involving fundamental differences in philosophy that move beyond mere
differences of opinion.

Like most controversies in which one group does not accept the
definitions, facts, or logic of another, the differences are unlikely
to be resolved by further discussion, no matter how sensible and
patient.   The material in this appendix is provided for the benefit
of those who cannot accept Section 3 or consider the discussion there
to be meaningless.

Independent of the details of the discussion above, in the case of
URNs, the IETF is faced with a pair of problems that are ultimately
faced sooner or later by all voluntary standards bodies: nothing
except quality and broad community consensus prevents a standard from
being ignored in the marketplace and nothing prevents another body
from creating a competing standard. The effort required to create a
competing standard can be increased and its potential for confusion
can be reduced somewhat by various measures -- measures the IETF has
rarely tried to actually use -- but those measures are rarely
effective when the other body is convinced that they have legitimate
and significant needs that differ from the original atandard.
Because of those problems, the key question for the URN effort is
ultimately not whether a clear enough distinction exists between
names and locator or location-based information, nor whether
"persistent" can be defined clearly enough, nor even whether the
communities and requirements described in Section 3 are valid or will
be judged valid in retrospect in a few decades or centuries.
Instead, the question is whether the IETF is willing to evolve and
adapt the URN definition to accommodate those perceived needs or
whether if prefers to have that work done elsewhere, either by
adoption in the broader community and marketplace of a different
approach or, potentially, even a competing URN standard. If, in the
long run, those other communities and perspectives turn out to be
wrong, the additional features will atrophy. But that would be true
whether they are specified and standardized in the IETF or elsewhere.

Appendix B. A Plausible Development Scenario

   NOTE IN DRAFT: this appendix is included in draft -01 to summarize
   some discussions on the mailing list in May and June 2014 for the
   convenience of the WG and possible discussions at IETF 90. It really
   is not part of this document and will be removed in the next version.

   The question has come up several times about what a URN syntax might
   look like when the URI (i.e., RFC 3986) constraints were removed and
   the questions about matching and resolution mechanisms that have
   plagued the WG were addressed.

One possible answer is that, if some of those questions can be successfully ignored, e.g., by never having the equivalent of query or fragment components treated as part of the URN for matching purposes and by using the same resolution framework for all URN, one could preserve the generic URI syntax, effectively just using this specification to break the link with some of the semantics specified in RFC 3986. That strategy makes sense only if the IETF is convinced that it understands all present and potential URNs well enough to specify those properties globally, possibly using an IANA registry for pointers to resolution mechanisms (at one per URN NID) for which see below.

If the community is less confident that it understands the full range of requirements for future URN namespaces, then one might, for example, generalize URNs and extend RFC 2141 so that a URN was, conceptually,

    "URN" NID NSS [ServiceRequests...]

It will ultimately make a difference whether "a URN" is the complete URN string as above or just urn:NID:NSS or, put differently, whether the ServiceRequests are part of the NSS. But it makes less difference in the near term than out trying to make general URIs work for URNs would suggest.

In the above, a ServiceRequest is, again conceptually, a tuple of

    ServiceType ComparisonIndicator ServiceTarget RequestParameters...

ServiceType is nominally some sort of keyword. ComparisonIndicator tells something trying to compare a pair of URNs for identity whether that particular ServiceRequest counts or should be ignored. ServiceTarget identifies where the ServiceRequest is to be sent and, depending on the ServiceType, may be a keyword indicator or, at the risk of descending into recursion hell, a URL or URN. And RequestParameters are anything the ServiceType definition says they are.

Any of those may be

o defined in the NID registration and omitted from (prohibited in) the URN string

o allowed by the NID registration but explicitly included in the URN string

o defined in the NID registration as a default but allowed in the URN string as an override

o prohibited entirely by the NID registration (effectively duplicating the "don't do that" rule of 2141 on a per-NID basis).

In addition, ServiceTarget might be specified in the NID registration to identify an IANA registry or domain subtree.

Presumably the NID registration may also specify whether anything not required is prohibited, and the other variations on that theme.

Requests/specifications for location information, assorted metadata, or model or actual objects themselves are then just specialized ServiceRequests. In particular, "Fragment" disappears as a special type of syntax and reappears as a Service Request that is applicable to some NIDs and not others and whose meaning and action (and how it is "resolved") are specified on an NID basis and as above. "Query" disappears too, not because it (or the syntax) are necessarily problematic but because the term itself is misleading for many possible types of ServiceRequests and therefore causes more confusion than it clears up.

ALmost independent of the above, unless we globally allow or prohibit non-ASCII content in URN strings, the registration/ definition of the NID would presumably identify what characters are permitted and, if necessary, how they are interpreted for matching purposes.

Appendix C. Change Log

RFC Editor: Please remove this appendix before publication.

Appendix C.1. Changes from version -00 to -01

   o Revised Section 1 slightly and added some new material to try to
      address questions raised on the mailing list.

   o Added Section 2, reflecting an email exchange.

   o Added a Security Considerations section, replacing the placeholder
      in the previous version.

   o Added Appendix Appendix A and inserted a note in Section 3
      pointing to it.

   o Added temporary Appendix Appendix B for this version only.

   o Enhanced and updated the Acknowledgments section.

   o The usual small clarifications and editorial changes.

Author's Address

   John C Klensin
   1770 Massachusetts Ave, Ste 322
   Cambridge, MA 02140
   USA

   Phone: +1 617 245 1457
   Email: john-ietf@jck.com