          Functional Requirements for Uniform Resource Names

Status of this Memo

   This memo provides information for the Internet community. This memo
   does not specify an Internet standard of any kind. Distribution of
   this memo is unlimited.

1. Introduction

   This document specifies a minimum set of requirements for a kind of
   Internet resource identifier known as Uniform Resource Names (URNs).
   URNs fit within a larger Internet information architecture, which in
   turn is composed of, additionally, Uniform Resource Characteristics
   (URCs), and Uniform Resource Locators (URLs). URNs are used for
   identification, URCs for including meta-information, and URLs for
   locating or finding resources. It is provided as a basis for
   evaluating standards for URNs. The discussions of this work have
   occurred on the mailing list uri@bunyip.com and at the URI Working
   Group sessions of the IETF.

   The requirements described here are not necessarily exhaustive; for
   example, there are several issues dealing with support for
   replication of resources and with security that have been discussed;
   however, the problems are not well enough understood at this time to
   include specific requirements in those areas here.

   Within the general area of distributed object systems design, there
   are many concepts and designs that are discussed under the general
   topic of "naming". The URN requirements here are for a facility that
   addresses a different (and, in general, more stringent) set of needs
   than are frequently the domain of general object naming.

   The requirements for Uniform Resource Names fit within the overall
   architecture of Uniform Resource Identification. In order to build
   applications in the most general case, the user must be able to
   discover and identify the information, objects, or what we will call
   in this architecture resources, on which the application is to
   operate. Beyond this statement, the URI architecture does not define
   "resource." As the network and interconnectivity grow, the ability
   to make use of remote, perhaps independently managed, resources will

become more and more important. This activity of discovering and
utilizing resources can be broken down into those activities where
one of the primary constraints is human utility and facility and
those in which human involvement is small or nonexistent. Human
naming must have such characteristics as being both mnemonic and
short. Humans, in contrast with computers, are good at heuristic
disambiguation and wide variability in structure. In order for
computer and network based systems to support global naming and
access to resources that have perhaps an indeterminate lifetime, the
flexibility and attendant unreliability of human-friendly names
should be translated into a naming infrastructure more appropriate
for the underlying support system. It is this underlying support
system that the Internet Information Infrastructure Architecture
(IIIA) is addressing.

Within the IIIA, several sorts of information about resources are
specified and divided among different sorts of structures, along
functional lines. In order to access information, one must be able
to discover or identify the particular information desired,
determined both how and where it might be used or accessed. The
partitioning of the functionality in this architecture is into
uniform resource names (URN), uniform resource characteristics (URC),
and uniform resource locators (URL). A URN identifies a resource or
unit of information. It may identify, for example, intellectual
content, a particular presentation of intellectual content, or
whatever a name assignment authority determines is a distinctly
namable entity. A URL identifies the location or a container for an
instance of a resource identified by a URN. The resource identified
by a URN may reside in one or more locations at any given time, may
move, or may not be available at all. Of course, not all resources
will move during their lifetimes, and not all resources, although
identifiable and identified by a URN will be instantiated at any
given time. As such a URL is identifying a place where a resource
may reside, or a container, as distinct from the resource itself
identified by the URN. A URC is a set of meta-level information
about a resource. Some examples of such meta-information are: owner,
encoding, access restrictions (perhaps for particular instances),
cost.

With this in mind, we can make the following statement:

o The purpose or function of a URN is to provide a globally unique,
   persistent identifier used for recognition, for access to
   characteristics of the resource or for access to the resource
   itself.

More specifically, there are two kinds of requirements on URNs: requirements on the functional capabilities of URNs, and requirements on the way URNs are encoded in data streams and written communications.

2. Requirements for functional capabilities

   These are the requirements for URNs' functional capabilities:

   o Global scope: A URN is a name with global scope which does not
     imply a location. It has the same meaning everywhere.

   o Global uniqueness: The same URN will never be assigned to two
     different resources.

   o Persistence: It is intended that the lifetime of a URN be
     permanent. That is, the URN will be globally unique forever, and
     may well be used as a reference to a resource well beyond the
     lifetime of the resource it identifies or of any naming authority
     involved in the assignment of its name.

   o Scalability: URNs can be assigned to any resource that might
     conceivably be available on the network, for hundreds of years.

   o Legacy support: The scheme must permit the support of existing
     legacy naming systems, insofar as they satisfy the other
     requirements described here. For example, ISBN numbers, ISO
     public identifiers, and UPC product codes seem to satisfy the
     functional requirements, and allow an embedding that satisfies
     the syntactic requirements described here.

   o Extensibility: Any scheme for URNs must permit future extensions to
     the scheme.

   o Independence: It is solely the responsibility of a name issuing
     authority to determine the conditions under which it will issue a
     name.

   o Resolution: A URN will not impede resolution (translation into a
     URL, q.v.). To be more specific, for URNs that have corresponding
     URLs, there must be some feasible mechanism to translate a URN to a
     URL.

3. Requirements for URN encoding

   In addition to requirements on the functional elements of the URNs,
   there are requirements for how they are encoded in a string:

o Single encoding: The encoding for presentation for people in clear
  text, electronic mail and the like is the same as the encoding in
  other transmissions.

o Simple comparison: A comparison algorithm for URNs is simple,
  local, and deterministic. That is, there is a single algorithm for
  comparing two URNs that does not require contacting any external
  server, is well specified and simple.

o Human transcribability: For URNs to be easily transcribable by
  humans without error, they should be short, use a minimum of
  special characters, and be case insensitive. (There is no strong
  requirement that it be easy for a human to generate or interpret a
  URN; explicit human-accessible semantics of the names is not a
  requirement.) For this reason, URN comparison is insensitive to
  case, and probably white space and some punctuation marks.

o Transport friendliness: A URN can be transported unmodified in the
  common Internet protocols, such as TCP, SMTP, FTP, Telnet, etc., as
  well as printed paper.

o Machine consumption: A URN can be parsed by a computer.

o Text recognition: The encoding of a URN should enhance the
  ability to find and parse URNs in free text.

4. Implications

   For a URN specification to be acceptible, it must meet the previous
   requirements. We draw a set of conclusions, listed below, from those
   requirements; a specification that satisfies the requirments without
   meetings these conclusions is deemed acceptable, although unlikely to
   occur.

o To satisfy the requirements of uniqueness and scalability, name
  assignment is delegated to naming authorities, who may then assign
  names directly or delegate that authority to sub-authorities.
  Uniqueness is guaranteed by requiring each naming authority to
  guarantee uniqueness. The names of the naming authorities
  themselves are persistent and globally unique and top level
  authorities will be centrally registered.

o Naming authorities that support scalable naming are encouraged, but
  not required. Scalability implies that a scheme for devising names
  may be scalable both at its terminators as well as within the
  structure; e.g., in a hierarchical naming scheme, a naming
  authority might have an extensible mechanism for adding new
  sub-registries.

   o It is strongly recommended that there be a mapping between the
     names generated by each naming authority and URLs. At any specific
     time there will be zero or more URLs into which a particular URN
     can be mapped. The naming authority itself need not provide the
     mapping from URN to URL.

   o For URNs to be transcribable and transported in mail, it is
     necessary to limit the character set usable in URNs, although there
     is not yet consensus on what the limit might be.

   In assigning names, a name assignment authority must abide by the
   preceding constraints, as well as defining its own criteria for
   determining the necessity or indication of a new name assignment.

5. Other considerations

   There are three issues about which this document has intentionally
   not taken a position, because it is believed that these are issues to
   be decided by local determination or other services within an
   information infrastructure. These issues are equality of resources,
   reflection of visible semantics in a URN, and name resolution.

   One of the ways in which naming authorities, the assigners of names,
   may choose to make themselves distinctive is by the algorithms by
   which they distinguish or do not distinguish resources from each
   other. For example, a publisher may choose to distinguish among
   multiple printings of a book, in which minor spelling and
   typographical mistakes have been made, but a library may prefer not
   to make that distinction. Furthermore, no one algorithm for testing
   for equality is likely to applicable to all sorts of information.
   For example, an algorithm based on testing the equality of two books
   is unlikely to be useful when testing the equality of two
   spreadsheets. Thus, although this document requires that any
   particular naming authority use one algorithm for determining whether
   two resources it is comparing are the same or different, each naming
   authority can use a different such algorithm and a naming authority
   may restrict the set of resources it chooses to identify in any way
   at all.

   A naming authority will also have some algorithm for actually
   choosing a name within its namespace. It may have an algorithm that
   actually embeds in some way some knowledge about the resource. In
   turn, that embedding may or may not be made public, and may or may
   not be visible to potential clients. For example, an unreflective
   URN, simply provides monotonically increasing serial numbers for
   resources. This conveys nothing other than the identity determined
   by the equality testing algorithm and an ordering of name assignment
   by this server. It carries no information about the resource itself.

An MD5 of the resource at some point, in and of itself may be
reflective of its contents, and, in fact, the naming authority may be
perfectly willing to publish the fact that it is using MD5, but if
the resource is mutable, it still will be the case that any potential
client cannot do much with the URN other than check for equality.
If, in contrast, a URN scheme has much in common with the assignment
ISBN numbers, the algorithm for assigning them is public and by
knowing it, given a particular ISBN number, one can learn something
more about the resource in question. This full range of
possibilities is allowed according to this requirements document,
although it is intended that naming authorities be discouraged from
making accessible to clients semantic information about the resource,
on the assumption that that may change with time and therefore it is
unwise to encourage people in any way to depend on that semantics
being valid.

Last, this document intentionally does not address the problem of
name resolution, other than to recommend that for each naming
authority a name translation mechanism exist. Naming authorities
assign names, while resolvers or location services of some sort
assist or provide URN to URL mapping. There may be one or many such
services for the resources named by a particular naming authority.
It may also be the case that there are generic ones providing service
for many resources of differing naming authorities. Some may be
authoritative and others not. Some may be highly reliable or highly
available or highly responsive to updates or highly focussed by other
criteria such as subject matter. Of course, it is also possible that
some naming authorities will also act as resolvers for the resources
they have named. This document supports and encourages third party
and distributed services in this area, and therefore intentionally
makes no statements about requirements of URNs or naming authorities
on resolvers.

Security Considerations

Applications that require translation from names to locations, and
the resources themselves may require the resources to be
authenticated. It seems generally that the information about the
authentication of either the name or the resource to which it refers
should be carried by separate information passed along with the URN
rather than in the URN itself.

Authors' Addresses

    Larry Masinter
    Xerox Palo Alto Research Center
    3333 Coyote Hill Road
    Palo Alto, CA 94304

    Phone: (415) 812-4365
    Fax: (415) 812-4333
    EMail: masinter@parc.xerox.com

    Karen Sollins
    MIT Laboratory for Computer Science
    545 Technology Square
    Cambridge, MA 02139

    Voice: (617) 253-6006
    Phone: (617) 253-2673
    EMail: sollins@lcs.mit.edu