

INTERNET-DRAFT
<[draft-masinter-url-process-01.txt](#)>
March 26, 1997
expires June 4, 1997

Larry Masinter
Dan Zigmond
Harald T. Alvestrand

Guidelines and Process for new URL Schemes

Status of this Memo

This document is an Internet-Draft. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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.''

To learn the current status of any Internet-Draft, please check the ``[1id-abstracts.txt](#)'' listing contained in the Internet-Drafts Shadow Directories on [ftp.is.co.za](#) (Africa), [nic.nordu.net](#) (Europe), [munnari.oz.au](#) (Pacific Rim), [ds.internic.net](#) (US East Coast), or [ftp.isi.edu](#) (US West Coast).

Issues:

Registration process isn't really there.

Abstract

A Uniform Resource Locator (URL) is a compact string representation of the location for a resource that is available via the Internet. [RFC URL-SYNTAX] defines the general syntax and semantics of URLs. This document provides guidelines for the definition of new URL schemes and describes the process by which they are registered.

1. Introduction

In addition to specifying the general syntax for Uniform Resource Locators, [RFC 1738](#) defined a number of generally useful URL schemes and promised that a mechanism for registering new schemes would be established. Several new URLs have been proposed since that time, but the procedure for standardizing these schemes has never been fully defined. This document describes the current practice and offers some guidance for authors of new schemes.

One process for defining URL schemes is via the Internet standards process: new URL schemes should be described in standards-track RFCs. The Internet Assigned Numbers Authority (IANA) maintains a registry of all URL schemes defined in this way.

2. Guidelines for new URL schemes

Because new URL schemes potentially complicate client software, new schemes must have demonstrable utility and operability, as well as compatibility with existing URL schemes. This section elaborates these criteria.

2.1 Syntactic compatibility

New URL schemes should follow the same syntactic conventions of existing schemes when appropriate.

2.1.1 Use of initial "://" for Internet host addresses

Many proposed new URL schemes seem to use "://" as a kind of indicator that what follows is a URL. However, the use of the top level "://" is indicative of an Internet host address, and not a top level marker.

2.1.2 Compatibility with relative URLs

URL schemes should use the generic-URL syntax if they are intended to be used with relative URLs. A description of the allowed relative forms should be included in the scheme's definition. Many applications use relative URLs extensively.

- o Can it be parsed according to RFC URL-SYNTAX - that is, if the tokens "://", "/", ";", "?" and "#" are used, do they have the meaning given in RFC URL-SYNTAX?
- o Does it make sense to use it in relative URLs like those RFC URL-SYNTAX specifies?
- o If something is designed to be broken into pieces, does it document what those pieces are, why it should be broken in this way, and why the breaks aren't where URL-SYNTAX says that they usually should be?
- o If it has a hierarchy, does it go left-to-right and with slash separators like RFC URL-SYNTAX? If not, why not?

2.1.3 Does it start with "ur"?

Any scheme starting with the letters "U" and "R", in particular if it attaches any of the meanings "uniform", "universal" or "unifying" to the first letter, is going to cause intense debate, and generate much heat (but maybe little light).

2.2 Is the scheme well defined?

It is important that the semantics of the "resource" that a URL "locates" be well defined. This might mean different things

depending on the nature of the URL scheme.

2.2.1 Clear mapping from other name spaces

In many cases, new URL schemes are defined as ways to translate other protocols and name spaces into the general framework of URLs. The "ftp" URL scheme translates from the FTP protocol, while the "mid" URL scheme translates from the Message-ID field of messages.

In either case, the description of the mapping must be complete, must describe how character sets get encoded or not in URLs, must describe exactly how all legal values of the base standard can be represented using the URL scheme, and exactly which modifiers, alternate forms and other artifacts from the base standards are included or not included. These requirements are elaborated below.

2.2.2 URL schemes associated with network protocols

Most new URL schemes are associated with network resources that have one or several network protocols that can access them. The 'ftp', 'news', and 'http' schemes are of this nature. For such schemes, the specification should completely describe how URLs are translated into protocol actions in sufficient detail to make the access of the network resource unambiguous. If an implementation of the URL scheme requires some configuration, the configuration elements must be clearly identified. (For example, the 'news' scheme, if implemented using NTTP, requires configuration of the NTTP server.)

2.2.3 Character encoding

When describing URL schemes in which (some of) the elements of the URL are actually representations of sequences of characters, care should be taken not to introduce unnecessary variety in the ways in which characters are encoded into octets and then into URL characters. Unless there is some compelling reason for a particular scheme to do otherwise, translating character sequences into UTF-8 [[RFC 2044](#)] and then subsequently using the %HH encoding for unsafe octets is recommended.

2.2.4 Definition of non-protocol URL schemes

In some cases, URL schemes do not have particular network protocols associated with them, because their use is limited to contexts where the access method is understood. This is the case, for example, with the "cid" and "mid" URL schemes. For these URL schemes, the specification should describe the notation of the scheme and a complete mapping of the locator from its source.

2.2.5 Definition of URL schemes not associated with data resources

Most URL schemes locate Internet resources that correspond to data objects that can be retrieved or modified. This is the case with "ftp" and "http", for example. However, some URL schemes do not; for example, the "mailto" URL scheme corresponds to an Internet mail address.

If a new URL scheme does not locate resources that are data objects, the properties of names in the new space must be clearly defined.

2.2.6 Definition of operations

In some contexts (for example, HTML forms) it is possible to specify any one of a list of operations to be performed on a specific URL. (Outside forms, it is generally assumed to be something you GET.)

The URL scheme definition should describe all well-defined operations on the URL identifier, and what they are supposed to do.

Some URL schemes (for example, "telnet") provide location information for hooking onto bidirectional data streams, and don't fit the "infoaccess" paradigm of most URLs very well; this should be documented.

NOTE: It is perfectly valid to say that "no operation apart from GET is defined for this URL". It is also valid to say that "there's only one operation defined for this URL, and it's not very GET-like". The important point is that what is defined on this type is described.

2.3 Demonstrated utility

URL schemes should have demonstrated utility. New URL schemes are expensive things to support. Often they require special code in browsers, proxies, and/or servers. Having a lot of ways to say the same thing needlessly complicates these programs without adding value to the Internet.

The kinds of things that are useful include:

- o Things that cannot be referred to in any other way.
- o Things where it is much easier to get at them using this scheme than (for instance) a proxy gateway.

2.3.1 Proxy into HTTP/HTML

One way to provide a demonstration of utility is via a gateway

which provides objects in the new scheme for clients using an existing protocol. It is much easier to deploy gateways to a new service than it is to deploy browsers that understand the new URL object.

Things to look for when thinking about a proxy are:

- o Is there a single global resolution mechanism whereby any proxy can find the referenced object?
- o If not, is there a way in which the user can find any object of this type, and "run his own proxy"?
- o Are the operations mappable one-to-one (or possibly using modifiers) to HTTP operations?
- o Is the type of returned objects well defined?
 - * as MIME content-types?
 - * as something that can be translated to HTML?
- o Is there running code for a proxy?

2.4 Are there security considerations?

Above and beyond the security considerations of the base mechanism a scheme builds upon, one must think of things that can happen in the normal course of URL usage.

In particular:

- o Does the user need to be warned that such a thing is happening without an explicit request (GET for the source of an IMG tag, for instance)? This has implications for the design of a proxy gateway, of course.
- o Is it possible to fake URLs of this type that point to different things in a dangerous way?
- o Are there mechanisms for identifying the requester that can be used or need to be used with this mechanism (the From: field in a mailto: URL, or the Kerberos login required for AFS access in the AFS: url, for instance)?
- o Does the mechanism contain passwords or other security information that are passed inside the referring document in the clear (as in the "ftp" URL, for instance)?

2.5 Does it start with UR?

Any scheme starting with the letters "U" and "R", in particular if it attaches any of the meanings "uniform", "universal" or "unifying" to the first letter, is going to cause intense debate, and generate much heat (but maybe little light).

Any such proposal should either make sure that there is a large consensus behind it that it will be the only scheme of its type, or

pick another name.

2.6 Non-considerations

Some issues that are often raised but are not relevant to new URL schemes include the following.

2.6.1 Is it an URL, an URN or something else?

This classification has proved interesting in theory, but not terribly useful when evaluating schemes.

2.6.2 Are all objects accessible?

Can all objects in the world that are validly identified by a scheme be accessed by any UA implementing it?

Sometimes the answer will be yes and sometimes no; often it will depend on factors (like firewalls or client configuration) not directly related to the scheme itself.

3. Revision process

NOTE: THIS SECTION IS ENTIRELY TBD. REVIEW COMMITTEE? PRIVATE URLS?

URL schemes will have either a standards track RFC, or else they will be a registration at IANA. where include the whole draft. URL schemes will have a review panel, appointed by IETF AD, who may not reject a URL scheme but who may provide a 2 sentence recommendation about the use of the URL scheme. Conflicting registrations are possible for non-standard URL schemes, and the order in the IANA list of conflicting registrations will be determined by a random number generator.

4. Security considerations

New URL schemes are required to address all security considerations in their definitions.

5. IANA considerations

This document requires IANA to register URL schemes according to the process outlined in [section 3](#).

6. References

[RFC2022] F. Yergeau, "UTF-8, A Transformation Format of Unicode and ISO 10646", Alis Technologies, October 1996.

[URL-SYNTAX]
Berners-Lee, Fielding, Masinter, "Uniform Resource Locators: Generic Syntax and Semantics", <[draft-fielding-url-syntax-04](#)>.

7. Author's Addresses

Larry Masinter
Xerox Corporation
Palo Alto Research Center
3333 Coyote Hill Road
Palo Alto, CA 94304
Fax: +1-415-812-4333
EMail: masinter@parc.xerox.com

Dan Zigmond
Wink Communications
1001 Marina Village Parkway
Alameda CA 94610
Fax: +1-510-337-2960
Phone: +1-510-337-6359
Email: dan.zigmond@wink.com

Harald T. Alvestrand
UNINETT A/S
Postboks 6683 Elgeseter 7002
Trondheim, Norway
Tel: +47 73 59 70 94
EMail: Harald.T.Alvestrand@uninett.no