

Definition of the URL  
MIME External-Body Access-Type

Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

1. Abstract

This memo defines a new access-type for message/external-body MIME parts for Uniform Resource Locators (URLs). URLs provide schemes to access external objects via a growing number of protocols, including HTTP, Gopher, and TELNET. An initial set of URL schemes are defined in [RFC 1738](#).

2. Introduction

The Multipurpose Internet Message Extensions (MIME) define a facility whereby an object can contain a reference or pointer to some form of data rather than the actual data itself. This facility is embodied in the message/external-body media type defined in [RFC 1521](#). Use of this facility is growing as a means of conserving bandwidth when large objects are sent to large mailing lists.

Each message/external-body reference must specify a mechanism whereby the actual data can be retrieved. These mechanisms are called access types, and [RFC 1521](#) defines an initial set of access types: "FTP", "ANON-FTP", "TFTP", "LOCAL-FILE", and "MAIL-SERVER".

Uniform Resource Locators, or URLs, also provide a means by which remote data can be retrieved automatically. Each URL string begins with a scheme specification, which in turn specifies how the remaining string is to be used in conjunction with some protocol to retrieve the data. However, URL schemes exist for protocol operations that have no corresponding MIME message/external-body access type. Registering an access type for URLs therefore provides message/external-body with access to the retrieval mechanisms of URLs that are not currently available as access types. It also provides access to any future mechanisms for which URL schemes are developed.

This access type is only intended for use with URLs that actually retrieve something. Other URL mechanisms, e.g. mailto, may not be used in this context.

### 3. Definition of the URL Access-Type

The URL access-type is defined as follows:

- (1) The name of the access-type is URL.
- (2) A new message/external-body content-type parameter is used to actually store the URL string. The name of the parameter is also "URL", and this parameter is mandatory for this access-type. The syntax and use of this parameter is specified in the next section.
- (3) The phantom body area of the message/external-body is not used and should be left blank.

For example, the following message illustrates how the URL access-type is used:

```
Content-type: message/external-body; access-type=URL;
              URL="http://www.foo.com/file"
```

```
Content-type: text/html
Content-Transfer-Encoding: binary
```

```
THIS IS NOT REALLY THE BODY!
```

### 3.1. Syntax and Use of the URL parameter

Using the ANBF notations and definitions of [RFC 822](#) and [RFC 1521](#), the syntax of the URL parameter is as follows:

URL-parameter := "<" URL-word \*( \*LWSP-char URL-word ) ">"

URL-word := token

          ; Must not exceed 40 characters in length

The syntax of an actual URL string is given in [RFC 1738](#). URL strings can be of any length and can contain arbitrary character content. This presents problems when URLs are embedded in MIME body part headers that are wrapped according to [RFC 822](#) rules. For this reason they are transformed into a URL-parameter for inclusion in a message/external-body content-type specification as follows:

- (1) A check is made to make sure that all occurrences of SPACE, CTLs, double quotes, backslashes, and 8-bit characters in the URL string are already encoded using the URL encoding scheme specified in [RFC 1738](#). Any unencoded occurrences of these characters must be encoded. Note that the result of this operation is nothing more than a different representation of the original URL.
- (2) The resulting URL string is broken up into substrings of 40 characters or less.
- (3) Each substring is placed in a URL-parameter string as a URL-word, separated by one or more spaces. Note that the enclosing quotes are always required since all URLs contain one or more colons, and colons are special characters [[RFC 1521](#)].

Extraction of the URL string from the URL-parameter is even simpler: The enclosing quotes and any linear whitespace are removed and the remaining material is the URL string.

The following example shows how a long URL is handled:

```
Content-type: message/external-body; access-type=URL;
              URL="ftp://ftp.deepdirs.org/1/2/3/4/5/6/7/
                  8/9/10/11/12/13/14/15/16/17/18/20/21/
                  file.html"
```

```
Content-type: text/html
Content-Transfer-Encoding: binary
```

THIS IS NOT REALLY THE BODY!

Some URLs may provide access to multiple versions of the same object in different formats. The HTTP URL mechanism has this capability, for example. However, applications may not expect to receive something whose type doesn't agree with that expressed in the message/external-body, and may in fact have already made irrevocable choices based on this information.

Due to these considerations, the following restriction is imposed: When URLs are used in the context of an access-type only those versions of an object whose content-type agrees with that specified by the inner message/external-body header can be retrieved and used.

#### 4. Security Considerations

The security considerations of using URLs in the context of a MIME access-type are no different from the concerns that arise from their use in other contexts. The specific security considerations associated with each type of URL are discussed in the URL's defining document.

Note that the Content-MD5 field can be used in conjunction with any message/external-body access-type to provide an integrity check. This insures that the referenced object really is what the message originator intended it to be. This is not a signature service and should not be confused with one, but nevertheless is quite useful in many situations.

#### 5. Acknowledgements

The authors are grateful for the feedback and review provided by John Beck and John Klensin.

## 6. References

[RFC-822]

Crocker, D., "Standard for the Format of ARPA Internet Text Messages", STD 11, [RFC 822](#), UDEL, August 1982.

[RFC-1521]

Borenstein, N. and N. Freed, "MIME (Multipurpose Internet Mail Extensions): Mechanisms for Specifying and Describing the Format of Internet Message Bodies", [RFC 1521](#), Bellcore, Innosoft, September, 1993.

[RFC-1590]

Postel, J., "Media Type Registration Procedure", [RFC 1590](#), USC/Information Sciences Institute, March 1994.

[RFC-1738]

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

## 7. Authors' Addresses

Ned Freed  
Innosoft International, Inc.  
1050 East Garvey Avenue South  
West Covina, CA 91790  
USA

Phone: +1 818 919 3600  
Fax: +1 818 919 3614  
EMail: ned@innosoft.com

Keith Moore  
Computer Science Dept.  
University of Tennessee  
107 Ayres Hall  
Knoxville, TN 37996-1301  
USA

EMail: moore@cs.utk.edu