PROPOSED SMPTE STANDARD

for Television —

Declarative Data Essence — Local Identifier (lid:) URI Scheme

Table of contents

1 Scope

This standard defines the lid: uniform resource identifier (URI), and describes how it is used to identify instances of resources, such as web pages and graphics files, that are transmitted through unidirectional means, such as a television broadcast.

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent edition of the standards indicated below.

IETF RFC 2396, Uniform Resource Identifiers (URI): Generic Syntax

ISO/IEC 11578:1996, Information Technology — Open Systems Interconnection — Remote Procedure Call (RPC), Annex A, Universal Unique Identifier

3 Introduction

Content resources delivered by a one-way broadcast must be identified, stored in a storage system used by the receiver as they are received, and referenced by a uniform scheme for access by applications and systems. Broadcast receivers may use different types of storage devices; therefore, content broadcasters and application developers need a standard syntax for resource storage and reference that does not depend on the specific device or directory syntax, such as the file: URI scheme. A lid: can be bound to a resource entity during authoring and distribution, and may be used to name a device-independent storage location for the entity. The lid: scheme is syntactically similar to the http: scheme, but it is not intended to resolve lid: identifiers to locations outside the broadcast stream or local storage system, as is the case for http: DNS and resource resolution.

The lid: URI scheme enables content creators to assign an authority value that is globally unique. The lid: scheme supports relative paths for resource retrieval, so the authority component can be separately identified in applications to allow relative path references similar to http: and other URI references.

A single lid: can be used to identify different resource instances over time, and will resolve in a receiver to the last instance received with an equivalent lid:. Appropriate lid: identifiers can reduce the storage of redundant instances of resources for better memory efficiency. Storage management for lid: entities is implementation specific and beyond the scope of this specification, but it can be assumed that memory will be finite, and so will the period of persistence of any lid: entity. Applications using lid: should be designed to handle the case where resources have been deleted over time due to storage limitations.

4 Definition of local identifier (lid:) URI scheme

The lid: URI scheme describes URI references consisting of a sequence of characters which are independent of their coding in octets in any particular character set. The lid: URI fully complies with IETF RFC 2396 except for the overloading of the authority field in the deprecated form.

The layout of the lid: URI follows the generic URI syntax:

lid://<userinfo>@<host>:<port><path>?<query>#<fragment>

Userinfo is an optional string that enables message ID syntax forms of the authority field and, in combination with the host field, complies with the mid: scheme syntax defined in IETF RFC 2392.

Host is a string whose root is a registered domain name or a uuid (ISO/IEC 11578) in string form. Note that the uuid form is deprecated and is intended to support past common practice.

Port is a string to allow syntactic compatibility with IETF RFC 2616 and has no semantic meaning.

Path is a slash-separated string of components identical to the http: scheme syntax as defined in IETF RFC 2616.

Query and fragment are content-type dependent strings compliant with IETF RFC 2396.

Relative path syntax, as described in section 3 of IETF RFC 2396 is also permitted syntactically, but must only be used in cases where there is a guaranteed mechanism to resolve the absolute path (i.e., the BASE URI is well defined). Practical delivery considerations may require that lid: identified resources be delivered on broadcast channels using absolute paths to enable real-time storage in sequence of resource arrival, but relative path resolution must be supported for lid: resource retrieval, assuming an application specifies the base of the URI by other means.

The following are examples of lid:s:

lid://xbc.com/EveningNews/11-March-01/Pacific/main.html

lid://4F4182C71C1FDD4BA0937A7EB7B8B4C1@mail.xbc.com

A deprecated form of usage is to permit host to be an encoded UUID (ISO/IEC 11578). While technically the UUID name space overlaps the domain name space, in practice a collision is entirely improbable. Examples of this deprecated form using UUID are:

lid://4F4182C71C1FDD4BA0937A7EB7B8B4C1/images/logo.gif

lid://4F4182C7-1C1F-DD4B-A093-7A7EB7B8B4C1/images/ logo.gif

The UUID is represented as an ASCII hex coding resulting in 32 characters. Note that two syntaxes are permitted — one with some specifically placed separating hyphens and one without (see the BNF definition below).

When different resources are received with matching lid:s, the most recently received resource should be referenced using that lid:. Thus a lid: may refer to different resources over time. One lid: URI can be assigned for all instances of a resource, or multiple unique URIs can be assigned, one for each instance of the resource.

5 Resolution rules

A lid: URI is used to label a resource. Certain parts of the URI are ignored for the purposes of comparison, when the lid: is used for retrieval, or to replace a previously transmitted resource with an equivalent URI. When testing for equivalence, the query and fragment identifiers (i.e., characters in the lid: including and following the first ? or # character) in a lid: URI are ignored. Notwithstanding, references using fragment and query identifiers may function in ways defined by the content type being referenced.

When comparing two URIs to decide if they match or not, a receiver should use a case-sensitive octet-by-octet comparison of the entire URIs, with these exceptions:

– A port that is empty or not given is equivalent to the default port for http:, which is 80;

– Comparisons of host names shall be case insensitive;

– Comparisons of scheme names shall be case insensitive;

– An empty abs_path is equivalent to an abs_path of /.

Characters other than those in the reserved and unsafe sets (see IETF RFC 2396) are equivalent to their % HEX HEX encoding. For example, the following three URIs are equivalent:

lid://abc.com:80/~smith/home.html

lid://ABC.com/%7Esmith/home.html

lid://ABC.com:/%7esmith/home.html

Unlike http:, a lid: URI is not locatable without more information and is thus URN-like in the generic definition in that a URN is associated to a resource and independent of the resource's location. Therefore, the details of resolution of the location of a lid: is application dependent.

6 Normalization and equivalence

In many cases, different URI strings may actually identify the same resource. For example, the host names used in the URL are case insensitive, so the URL <lid://www.XBC.com> is equivalent to <lid://www.xbc.com>. In general, the rules for equivalence and definition of a normal form, if any, are scheme dependent. When a scheme uses elements of the common syntax, it will also use the common syntax equivalence rules; namely, that the scheme and hostname are case insensitive and a URL with an explicit :port, where the port is the default for the scheme, is equivalent to one where the port is elided.

7 Local identifier syntax BNF

The collected BNF for lid: URIs is as follows:

lid = "lid" ":" "//" authority [abs_path] ["?" query]
["#" fragment]

authority = server | uuid

server = as defined in RFC 2396

abs_path = as defined in RFC 2396

query = as defined in RFC 2396

fragment = as defined in RFC 2396

uuid = uuid_simple | uuid_idl

uuid_simple = 32hex

uuid_idl = 8hex "–" 4hex "–" 4hex "–" 4hex "–"
12hex

hex = as defined in RFC 2396

NOTE – The notation <n> (element) means exactly <n> occurrences of (element); e.g., 32hex means exactly 32 hex digits. Use of the uuid authority element is deprecated.

8 Security considerations

The local identifier URI scheme is subject to the same security implications as in general URI schemes, so the usual precautions apply. This means that some local identifier URIs may refer to resources that are not available (because they have not been received, for example), or to resources that have been received but were intentionally misidentified. The security issues associated with this mislabeling, as well as the security issues associated with the use of HTML content which is broadcast, are the same as those identified in section 11.1 of IETF RFC 2387.

Appropriate security mechanisms should be used in the delivery of content identified by lid: URIs. These include protection of the broadcast signal by data encryption and conditional access methods, and protection of content prior to broadcast so that invalid lid:s are not created, and valid lid:s are not modified.

Annex A (informative)
Converting other URI schemes to lid:

URL references using schemes such as http:, ftp:, and file: can be converted to valid lid:s by changing the scheme component and using the original name, host, path, fragment, and query components. This is useful, for instance, to deliver resources stored on Internet servers over a broadcast channel. Note that the original URL port and password fields have no semantic definition in lid:.

Example:

An Internet resident resource at the location

http://www.xbc.com/tv/text.txt

could be packaged in a broadcast stream with a header containing the resource identifier

lid://www.xbc.com/tv/text.txt

and that resource identifier could be used to store the text.txt entity in memory with a derived directory entry, which would be matched by the following lid: reference in an HTML document

href=lid://www.xbc.com/tv/text.txt?ID=myProgram

Annex B (informative)
Bibliography

IETF RFC 2387, The MIME Multipart/Related Content-Type

IETF RFC 2392, Content-ID and Message-ID Uniform Resource Locators

IETF RFC 2616, Hypertext Transfer Protocol — HTTP/1.1

IETF RFC 2717, Registration Procedures for URL Scheme Names

IETF RFC 2718, Guidelines for New URL Schemes