

Network Working Group	L. Masinter
Internet-Draft	Adobe
Intended status: Informational	July 12, 2009
Expires: January 13, 2010	

The "tdb" URI scheme: denoting described resources draft-masinter-dated-uri-06

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

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."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on January 13, 2010.

Copyright Notice

Copyright (c) 2009 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents in effect on the date of publication of this document (<http://trustee.ietf.org/license-info>). Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

Abstract

This document defines a URI scheme, "tdb" (standing for "Thing Described By"). It provides a semantic hook for allowing anyone at any time to mint a URI for anything that they can describe. Such URIs may include a timestamp to fix the description at a given date or time.

This URI scheme may reduce the need to define new URN namespaces merely for the purpose of creating stable identifiers. In addition, they provide a ready means for identifying "non-information resources" by semantic indirection -- a way of creating a URI for anything.

Note

This document is not a product of any working group. Many of the ideas here have been discussed since 2001. This document has been discussed on the mailing list <uri@w3.org>. Previous versions have couched "tdb" as a URN namespace, and included a "duri" scheme for fixing date without indirection, which seems unnecessary. It was originally written as a thought experiment as a way of resolving the use/mention problem in semantic web applications, but may have other uses.

Table of Contents

- 1. Overview and Requirements**
 - 1.1. Easy assignment of permanent identifiers**
 - 1.2. Persistent identifiers**
 - 1.3. URIs for abstractions**
- 2. Syntax**
- 3. Semantics**
- 4. Use as a Locator**
- 5. Hierarchy**
- 6. Timestamps in tdb URIs**
- 7. Additional Considerations**
 - 7.1. URI schemes for the description resource**
 - 7.2. Useful timestamps**
 - 7.3. Free assignment**
 - 7.4. Resolution**
 - 7.5. Why Names with Semantics?**
 - 7.6. Avoiding MetaData**
 - 7.7. Avoiding tdb**
 - 7.8. tdb and levels of indirection**
- 8. URI Specification Template**
- 9. IANA considerations**
- 10. Security Considerations**
- 11. Acknowledgements**
- 12. References**
 - 12.1. Normative References**
 - 12.2. Informative References**
- § Author's Address**

TOC

1. Overview and Requirements

The tdb URI scheme here solves several related problems:

TOC

1.1. Easy assignment of permanent identifiers

The URN specification [\[RFC1737\]](#) allows for many URN namespaces, and many have been registered. However, obtaining an appropriate URN in any of the currently defined URN namespaces may be difficult: a number of URN namespace registrations have been accompanied by comments that no other URN namespace was available for the class of documents for which identifiers were wanted.

TOC

1.2. Persistent identifiers

[\[RFC1737\]](#) defines several requirements for Uniform Resource Names. In particular, it requires "persistence":

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.

Many people have wondered how to create globally unique and persistent identifiers. There are a number of URI schemes and URN namespaces already registered. However, an absolute guarantee of both uniqueness and persistence is very difficult.

In some cases, the guarantee of persistence comes through a promise of good management practice, such as is encouraged in "**Cool URLs don't change**" [COOL]. However, relying on promise of good management practice is not the same as having a design that guarantees reliability independent of actual administrative practice.

A primary design goal for URIs is that they are intended to mean the same thing, no matter in what context they appear: a "Uniform" way to Identify a Resource. However, even when URIs have Uniform meaning from the point of view of the source of the reference, they don't guarantee stability over time. Despite best efforts and intentions, identifying information can change in unpredictable ways: domain names can disappear or be reassigned, name assigning organizations can change structure, responsibility, disappear, merge, or change in unpredictable ways.

There is a significant dependence in the interpretation of many URNs with the concept of "naming authority". The authority is presumably some individual or organization both to insure uniqueness of assignment and also to help with understanding the meaning of the link between the name and the named.

However, authorities, whether individuals or organizations, have a lifetime, and must be consulted at some point to understand the bindings. The functioning of names as unique identifiers and holders of meaning depends on having a reliable infrastructure of consulting the authority or the authorities records to determine the thing referenced.

1.3. URIs for abstractions

The description of URIs [RFC3986] describes a range for 'Resource' that is quite broad:

This specification does not limit the scope of what might be a resource; rather, the term "resource" is used in a general sense for whatever might be identified by a URI. Familiar examples include an electronic document, an image, a source of information with a consistent purpose (e.g., "today's weather report for Los Angeles"), a service (e.g., an HTTP-to-SMS gateway), and a collection of other resources. A resource is not necessarily accessible via the Internet; e.g., human beings, corporations, and bound books in a library can also be resources. Likewise, abstract concepts can be resources, such as the operators and operands of a mathematical equation, the types of a relationship (e.g., "parent" or "employee"), or numeric values (e.g., zero, one, and infinity).

One might use a URI such as "mailto:" email address to identify a person, or a "http:" URI to identify an abstract comment. However, this leaves the question of how one might identify, within the same context, both the system mailbox and the person to which it is assigned, or the web page at a http URI and the concept it describes. The "tdb" URI scheme allows ready assignment of URIs for abstractions that are distinguished from the media content that describes them.

The goal, then, of the "tdb" URI scheme is to provide a mechanism which is, at the same time:

permanent: The identity of the resource identified is not subject to reinterpretation over time.

explicitly bound: The mechanism by which the identified resource can be determined is explicitly included in the URI.

useful for non-networked items: Allows identification of resources outside the network: people, organizations, abstract concepts.

no administration: The mechanism does not depend on reliable administrative processes of authorities for either assignment or interpretation.

TOC

2. Syntax

A tdb URI takes the form:

```
duri:<timestamp>:<URI>
```

Where <timestamp> is a sequence of digits representing a date and time ([Section 6](#)) and <URI> is any valid URI.

TOC

3. Semantics

The tdb URI scheme is intended to be useful for describing entities, concepts, abstractions, and other items which may not themselves be network accessible resources, but have been at some point described by network accessible resources.

The meaning of a duri is "the resource (or fragment) that was identified by the <encoded-URI> (after hex decoding) at the very last instant of the date(time) given".

The intent is to use the inversion of "is a document about". It is common practice to give a reference for a concept by including a pointer to a document, segment, phrase that defines the concept. "tdb" attempts to capture this practice in URI space.

For example, one might use "tdb:2008:http://www.ietf.org" as a persistent identifier for the Internet Engineering Task Force, as described by the "http://www.ietf.org" as of the very last instant of the year 2008.

The "tdb" namespace differs from the URN methods for identifying abstractions because the designation of what is actually identified by the tdb doesn't depend on knowing the intention of the "assigner" of the identifier. Unlike "tag", "info", "cid", "mid" or related schemes, the identification is not dependent on the context of use.

The "tdb" scheme can be thought of as adding a level of semantic indirection to URI resolution.

TOC

4. Use as a Locator

A tdb URI is not a resource locator in a practical sense. It allows one to know that a resource was described at some point in time, but whether the description is still available, or whether that description is still meaningful, is ambiguous.

TOC

5. Hierarchy

The "thing described by" a network resource may bear little relationship to the "thing described by" a

relative pointer, so the "tdb" URI scheme seems to have no use cases for using "/" as a hierarchical delimiter.

6. Timestamps in tdb URIs

It is traditional in convention references and citations in printed works to include the date of publication; this practice serves the important purpose that the context of the naming can be determined.

While one could imagine using tdb without a timestamp, it would leave the possibility that a reference that is unambiguous at one time might become ambiguous at some other time. There are two ways that the date is useful for "tdb": it fixes the time of access of the resource, for variable descriptions, and it fixes the time of interpretation, for descriptions whose meaning (in natural language) might vary.

A timestamp SHOULD be supplied, since the network resources which provide descriptions can also change over time. The timestamp is allowed to be quite broad -- only a year -- or with as much precision as needed. This keeps "tdb" URIs relatively short. To avoid ambiguity, a single instant has been chosen -- for tdb this is "the last possible instant of the indicated range".

A timestamp in the tdb scheme is a simple expression of date, optional time, with arbitrary precision. The goal is to allow relatively short expressions with no ambiguity, but also with arbitrary precision. (Other date formats were considered, but arbitrary precision syntactic simplicity of only using digits time zones not.)

```
date = [ year [ month [ day [ hour [ minute [ second [ fraction ]]]]]]]
year   = 4digit
month  = 2digit
day    = 2digit
hour   = 2digit
minute = 2digit
second = 2digit
fraction = *digit
```

The representation of a date or time refers to the (open interval) instant just before the end of the given date/time range at the resolution supplied. 199912 is "just before" 1999, but 19991231 falls between them. If necessary, timestamps can include times and even fractional times, so that a generator of tdb's can be arbitrarily precise.

Timestamps are interpreted relative to International Atomic Time (TAI) [TAI]. The syntax and semantics are similar to those in [RFC2550]; in particular, using TAI avoids ambiguity about time zones and difficulties with leap seconds.

There are actually two dates to consider, with "tdb". There is the date that the resource is obtained, and there is the date that the description it makes is read, understood, and used to denote. Normally in a literary work in natural language which makes a reference to another work, both the reference itself and the work referenced are dated, e.g., a footnote in an article written in 1967 might talk about a "private communication" which itself had a date. The difference between a URI and a conventional literary reference is the desire to be able to extract the URI from its context and still retain its meaning.

7. Additional Considerations

7.1. URI schemes for the description resource

The "tdb" scheme is intended for use with resources which have retrievable resources that describe something else -- these "description resources" are intended as "information resources".

For example, use with a "http" URI can be used to refer to the subject of a web page (at it was described at the given time.) This can be a way of referring to a web site at some time in the past, or an organization that has changed, merged, split, or disappeared.

Local systems that have known-to-be unique host names can use "file" URIs with "tdb", for example,

```
tdb:20010814142327:file://this.example.com/c|/temp/test.txt
```

since this use is primarily focused on providing a unique way of identifying an abstraction, even if the referent of the abstraction is not widely known. (Using 'file:' URIs in this way without a fully qualified domain name would not be appropriate, because the interpretation is not uniform.)

One might consider using "tdb" with "data" to designate concepts that can be described uniquely briefly inline. For example,

```
tdb:2001:data: ,The%20US%20president
```

names the concept described by the (text/plain) string "The US president" at the very last instant of 2001. Of course, this practice is only useful if the referent of the data is (or was at the time) completely unique. Since "data" does not contain a way to designate content-language, the string in question would have to not be ambiguous as to its language. In the case of 'data', there is no assigning authority at all; the interpretation of the 'tdb' depend on the interpreting community.

Many URIs identify resources which do not clearly describe anything at all. The "home page" for an organization isn't nearly as good a resource to use to describe an organization as the organization's "about" page. But it is up to the minter of the tdb URI to choose wisely.

7.2. Useful timestamps

Timestamps far in the future are suspect, because the future content of a description resource cannot usually reliably predicted. Timestamps which precede the availability of the description resource should not be used either. For example, using a http URI with a timestamp before the description resource is also not recommended.

However, although these practices are not recommended, there is no assurance that they haven't been used; by itself, a tdb does not constitute an assertion that the description resource was available or assigned at the date specified.

Note that the use of the "very last instant" allows for the conventional bibliographic convention that a work published in 2009 can use "2009" as the date string, to refer to the work in the year of publication.

7.3. Free assignment

Because of the many possible schemes that can be used in the <URI> portion, there should be no

difficulty in almost any computational process being able to assign tdb's at will. Of course, it is necessary for there to be some resource which is available at some point in time, and to have a clock which is accurate to the granularity of the frequency of assignment.

TOC

7.4. Resolution

There are no resolution servers or processes for tdb URI. However, a tdb URI might be "resolvable" in the sense that a resource that was accessed at a point in time might have the result of that access cached or archived in an Internet archive service. See, for example, the "Internet Archive" project [\[archive\]](#). And the "tdb" is "resolvable" in the sense that the description resource can be accessed and interpreted.

TOC

7.5. Why Names with Semantics?

There are a number of URI and URN schemes that create otherwise unbound "names", where the scheme only provides for uniqueness, with some other agent or process or context providing the authority to interpret the meaning of the identifier at some point in the future. "tdb" is different, in that it is the agreement between the describer (the agent creating the tdb URI) and the receiver of the URI (the agent interpreting the tdb URI) to agree upon the semantics without any reference to any third party.

TOC

7.6. Avoiding MetaData

One might consider the date in a tdb URI to be just one piece of additional metadata about the URI, and consider adding other pieces of metadata as annotation.

However, the use of the date in a tdb URI is intended primarily as a mechanism of accomplishing uniqueness over time. No other bit of metadata or description readily fills that purpose. Further, the date is not descriptive (an assertion about the URI) but merely refining.

TOC

7.7. Avoiding tdb

Many applications of URIs already provide a context of timestamp. For example, one could imagine a hypertext system where the URIs contained within a document were intended to refer to the resources as of the date of the enclosing document. This would be a reasonable interpretation of URIs within an Internet archive system, for example.

And some applications of URIs arguably already contain the level of interpretive indirection that is explicit with "tdb". For example, one might consider the use of URIs as namespace names within XML [\[namespaces\]](#) as a reference to the "thing described by" the URI used.

TOC

7.8. tdb and levels of indirection

The "tdb" scheme introduces a level of semantic indirection. The puzzles and confusions about use and mention, name and reference, and levels of indirection have been puzzling and amusing for quite a while.

"It's long," said the Knight, "but it's very, very beautiful. Everybody that hears me sing it--either it brings tears into their eyes, or else--"

"Or else what?" said Alice, for the Knight had made a sudden pause.

"Or else it doesn't, you know. The name of the song is called 'Haddock's Eyes.'"

"Oh, that's the name of the song, is it?" Alice said, trying to feel interested.

"No, you don't understand," the knight said, looking a little vexed. "That's what the name is called. The name really is 'The Aged Aged Man.'"

"Then I ought to have said 'That's what the song is called?'" Alice corrected herself.

"No, you oughtn't: that's quite another thing! The song is called 'Ways and Means': but that's only what it's called, you know!"

"Well, what is the song, then?" said Alice, who was by this time completely bewildered.

"I was coming to that," the Knight said. "The song really is 'A-sitting On A Gate': and the tune's my own invention." **[LOOK]**

8. URI Specification Template

URI scheme name:

tdb

Status:

permanent

URI scheme syntax:

Briefly, the syntax is tdb:<date>:<URI>

The syntax is described in this document.

URI scheme semantics:

Semantic indirection at indicated date. Semantics are described in detail in this document.

Encoding considerations:

tdb URIs consist of a prefix followed by another URI, and should have the same encoding considerations as others.

Applications/protocols that use this URI scheme name:

This scheme was designed to resolve some of the use/mention ambiguities in semantic web applications that wish to "denote" concepts and other ideas and not just access resources over the Internet.

Interoperability considerations:

Existing semantic web applications may have other means of fixing meaning at a particular time or semantic indirection, but this should not in itself cause interoperability difficulties.

Security considerations:

See **Section 10** of this document.

Contact:

Larry Masinter tdb:2009:http://larry.masinter.net

Author/Change controller:

as above

References:

See References of this document.

9. IANA considerations

This document includes a URI scheme registration (**Section 8** that should be entered into the IANA registry of URI schemes as a permanent registration (once approved.)

10. Security Considerations

"tdb" identifiers are not any more reliable because they have dates. URIs don't contain enough information to supply the authority for deciding what was or wasn't at a given URI at a given date.

11. Acknowledgements

There have been many discussions over several years on the relationship of URLs, URNs, URIs, resources and resource identifiers, with many contributions. Particular thanks to Al Gilman, Aaron Swartz, Brian McBride, Stuart Williams, Michael Mealling, Ray Denenberg and Pat Hayes.

12. References

12.1. Normative References

- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "[Uniform Resource Identifiers \(URI\): Generic Syntax](#)," RFC 3986, January 2005.
- [TAI] Bureau International des Poids et Mesures, "International Atomic Time."
- [namespaces] Bray, T., Hollander, D., and A. Layman, "[Namespaces in XML](#)," W3C Recommendation REC-xml-names, January 1999.

12.2. Informative References

- [COOL] Berners-Lee, T., "[Cool URIs don't change](#)," 1998.
- [LOOK] Carroll, L., "[Through the Looking Glass](#)," 1872.
- [RFC1737] Sollins, K., "[Functional Requirements for Uniform Resource Names](#)," RFC 1737, December 1994.
- [RFC2550] Glassman, S., Manasse, M., and J. Mogul, "[Y10K and Beyond](#)," RFC 2550, April 1 1999.
- [archive] Kahle, B., "[Preserving the Internet](#)," Scientific American , March 1997.

Author's Address

Larry Masinter
Adobe
345 Park Ave
San Jose, CA 95110
US
Phone: +1 408 536 3024
Email: LMM@acm.org
URI: <http://larry.masinter.net>