Web Linking
draft-nottingham-http-link-header-06

Status of this Memo

   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

Abstract

This document specifies relation types for Web links, and defines a
registry for them. It also defines how to send such links in HTTP
headers with the Link header-field.

Table of Contents

1. Introduction

A means of indicating the relationships between resources on the Web,
as well as indicating the type of those relationships, has been
available for some time in HTML [W3C.REC-html401-19991224], and more
recently in Atom [RFC4287]. These mechanisms, although conceptually
similar, are separately specified. However, links between resources
need not be format-specific; it can be useful to have typed links
that are independent of their serialisation, especially when a
resource has representations in multiple formats.

To this end, this document defines a framework for typed links that
isn't specific to a particular serialisation. It does so by re-
defining the link relation registry established by Atom to have a
broader scope, and adding to it the relations that are defined by
HTML.

Furthermore, an HTTP header-field for conveying typed links was
defined in [RFC2068], but removed from [RFC2616], due to a lack of
implementation experience. Since then, it has been implemented in
some User-Agents (e.g., for stylesheets), and several additional use
cases have surfaced.

Because it was removed, the status of the Link header is unclear,
leading some to consider minting new application-specific HTTP
headers instead of reusing it. This document addresses this by re-
specifying the Link header as one such serialisation, with updated
but backwards-compatible syntax.

[[Feedback is welcome on the ietf-http-wg@w3.org mailing list,
although this is NOT a work item of the HTTPBIS WG.]]

2. Notational Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
"SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
document are to be interpreted as described in BCP 14, [RFC2119], as
scoped to those conformance targets.

This document uses the Augmented Backus-Naur Form (ABNF) notation of
[RFC2616], and explicitly includes the following rules from it:
quoted-string, token, SP (space). Additionally, the following rules
are included from [RFC3986]: URI and URI-Reference, and from
[RFC4288]: type-name and subtype-name.

3. **Links**

   In this specification, a link is a typed connection between two
   resources that are identified by IRIs [RFC3987], and is comprised of:
   o A context IRI, and
   o a link relation type (Section 4), and
   o a target IRI, and
   o optionally, target attributes.

   A link can be viewed as a statement of the form "{context IRI} has a
   {relation type} resource at {target IRI}, which has {target
   attributes}."

   Note that in the common case, the context IRI will also be a URI
   [RFC3986], because many protocols (such as HTTP) do not support
   dereferencing IRIs. Likewise, the target IRI will be converted to a
   URI (see [RFC3987], Section 3.1) in serialisations that do not
   support IRIs (e.g., the Link header).

   This specification does not place restrictions on the cardinality of
   links; there can be multiple links from and to a particular IRI, and
   multiple links of different types between two given IRIs. Likewise,
   the relative ordering of links in any particular serialisation, or
   between serialisations (e.g., the Link header and in-content links)
   is not specified or significant in this specification; applications
   that wish to consider ordering significant MAY do so.

   Target attributes are a set of key/value pairs that describe the link
   or its target; for example, a media type hint. This specification
   does not attempt to coordinate their names or use, but does provide
   common target attributes for use in the Link HTTP header.

   Finally, this specification does not define a general syntax for
   expressing links, nor mandate a specific context for any given link;
   it is expected that serialisations of links will specify both
   aspects. One such serialisation is communication of links through
   HTTP headers, specified in Section 5.

4. **Link Relation Types**

   A link relation type identifies the semantics of a link. For
   example, a link with the relation type "copyright" indicates that the
   resource identified by the target IRI is a statement of the copyright
   terms applying to the current context IRI.

   Relation types are not to be confused with media types [RFC4288];
   they do not identify the format of the representation that results

   when the link is dereferenced. Rather, they only describe how the
   current context is related to another resource.

   As such, relation types are not format-specific, and MUST NOT specify
   a particular format or media type that they are to be used with.
   Likewise, the context IRI for a given link is usually determined by
   the serialisation of the link (e.g., the Link header, a HTML
   document, etc.); a relation type SHOULD NOT specify the context IRI.

   Relation types SHOULD NOT infer any additional semantics based upon
   the presence or absence of another link relation, or its own
   cardinality of occurrence. An exception to this is the combination
   of the "alternate" and "stylesheet" registered relation types, which
   have special meaning in HTML4 for historical reasons.

   Consuming implementations SHOULD ignore relation types that they do
   not understand or have no need to process.

   There are two kinds of relation types: registered and extension.

4.1. Registered Relation Types

   Commonly-used relation types with a clear meaning that are shared
   across applications can be registered as tokens for convenience and
   to promote reuse. For example, "self" and "alternate" are registered
   relation types, because they are broadly useful.

   This specification establishes an IANA registry of such relation
   types; see Section 6.2.

   Registered relation types MUST conform to the token rule, and SHOULD
   conform to the sgml-name rule for compatibility with deployed
   implementations;

     sgml-name   = ALPHA *(ALPHA | DIGIT | "." | "-")

   Names that differ only in case from existing entries (e.g., "Foo" and
   "foo") MUST NOT be registered. Registered relation types MUST be
   compared character-by-character in a case-insensitive fashion.

   Although registered relation types are specified as tokens,
   applications wishing to internally refer to one using a URI MAY do so
   by considering it relative to the base URI
   "http://www.iana.org/assignments/relation/". However, the URI form
   of a registered relation type SHOULD NOT be serialised when an
   application specifies the use of a relation type, because a consuming
   implementation may not recognise it.

4.2. Extension Relation Types

   Applications that don't merit a registered relation type may use an
   extension relation type, which is a URI [RFC3986] that uniquely
   identifies the relation type. Although the URI MAY point to a
   resource that contains a definition of the semantics of the relation
   type, clients SHOULD NOT automatically access that resource to avoid
   overburdening its server.

   When extension relation types are compared, they MUST be compared as
   URIs in a case-sensitive fashion, character-by-character.

   Note that while extension relation types are required to be URIs, a
   serialisation of links MAY specify that they are expressed in another
   form, as long as they can be converted to URIs.

5. The Link Header Field

   The Link entity-header field provides a means for serialising one or
   more links in HTTP headers. It is semantically equivalent to the
   <LINK> element in HTML, as well as the atom:link feed-level element
   in Atom [RFC4287].

```
  Link               = "Link" ":" #link-value
  link-value         = "<" URI-Reference ">" *( ";" link-param )
  link-param         = ( ( "rel" "=" relation-types )
                     | ( "rev" "=" relation-types )
                     | ( "type" "=" type-name "/" subtype-name )
                     | ( "title" "=" quoted-string )
                     | ( "title*" "=" enc2231-string )
                     | ( "anchor" "=" <"> URI-Reference <"> )
                     | ( link-extension ) )
  link-extension     = token [ "=" ( token | quoted-string ) ]
  enc2231-string     = <extended-value, see [RFC2231], Section 7>
  relation-types     = relation-type |
                       <"> relation-type *( SP relation-type ) <">
  relation-type      = reg-relation-type | ext-relation-type
  reg-relation-type = token
  ext-relation-type = URI
```

   Each link-value conveys one target IRI as a URI-Reference (after
   conversion to one, if necessary; see [RFC3987], Section 3.1) inside
   angle brackets ("<>"). If the URI-Reference is relative, it MUST be
   resolved as per [RFC3986], Section 5. Note that any base IRI from
   the body's content is not applied.

   By default, the context of a link conveyed in the Link header field

is the IRI of the requested resource. When present, the anchor
parameter overrides this with another URI, such as a fragment of this
resource, or a third resource (i.e., when the anchor value is an
absolute URI). If the anchor parameter's value is a relative URI, it
MUST be resolved as per [RFC3986], Section 5. Note that any base URI
from the body's content is not applied.

Normally, the relation type of a link is conveyed in the "rel"
parameter's value. The "rev" parameter has also been used for this
purpose historically by some formats, and is included here for
compatibility with those uses, but its use is not encouraged nor
defined by this specification.

Note that extension relation types are REQUIRED to be absolute URIs
in Link headers, and MUST be quoted if they contain a semicolon (";")
or comma (",").

The "title", "title*", "type" and any link-extension link-params are
considered to be the target parameters for the link.

The "title" parameter is used to label the destination of a link such
that it can be used as a human-readable identifier (e.g. a menu
entry). Alternately, the "title*" parameter MAY be used encode this
label in a different character set, and/or contain language
information as per [RFC2231]. When using the enc2231-string syntax,
producers MUST NOT use a charset value other than 'ISO-8859-1' or
'UTF-8'.

The "type" parameter, when present, is a hint indicating what the
media type of the result of dereferencing the link should be. Note
that this is only a hint; for example, it does not override the
Content-Type header of a HTTP response obtained by following the
link.

5.1. Examples

NOTE: Non-ASCII characters used in prose for examples are encoded
using the format "Backslash-U with Delimiters", defined in Section
5.1 of [RFC5137].

For example:

Link: <http://example.com/TheBook/chapter2>; rel="previous";
      title="previous chapter"

indicates that "chapter2" is previous to this resource in a logical
navigation path.

Similarly,

    Link: </>; rel="http://example.net/foo"

indicates that the root resource ("/") has the extension relation
"http://example.net/foo".

The example below shows an instance of the Link header encoding
multiple links, and also the use of RFC 2231 encoding to encode both
non-ASCII characters and language information.

    Link: </TheBook/chapter2>;
          rel="previous"; title*=UTF-8'de'letztes%20Kapitel",
          </TheBook/chapter4>;
          rel="next"; title*=UTF-8'de'n%c3%a4chstes%20Kapitel"

Here, the second link has a title encoded in UTF-8, uses the German
language ("de"), and contains the Unicode code point U+00E4 ("LATIN
SMALL LETTER A WITH DIAERESIS").

Note that link-values may convey multiple links between the same
target and context IRIs; for example:

      Link: <http://example.org/>; rel=index;
            rel="start http://example.net/relation/other"

Here, the link to "http://example.org/" has the registered relation
types "index" and "start", and the extension relation type
"http://example.net/relation/other".

6. IANA Considerations

6.1. Link Header Registration

This specification updates the Message Header Registry entry for
"Link" in HTTP [RFC3864] to refer to this document.

Header field: Link
Applicable protocol: http
Status: standard
Author/change controller:
    IETF (iesg@ietf.org)
    Internet Engineering Task Force
Specification document(s):
  [this document]

6.2. Link Relation Type Registry

   This specification establishes the Link Relation Type Registry, and
   updates Atom [RFC4287] to refer to it in place of the "Registry of
   Link Relations".

   The requirements for registered relation types are described in
   Section 4.1.

   Relation types are registered on the advice of a Designated Expert
   (appointed by the IESG or their delegate), with a Specification
   Required (using terminology from [RFC5226]).

   Registration requests consist of the completed registration template
   below, typically published in an RFC or Open Standard (in the sense
   described by [RFC2026], Section 7). However, to allow for the
   allocation of values prior to publication, the Designated Expert may
   approve registration once they are satisfied that an RFC (or other
   Open Standard) will be published.

   The registration template is:

   o Relation Name:
   o Description:
   o Reference:
   o Notes: [optional]

   Upon receiving a registration request (usually via IANA), the
   Designated Expert should request review and comment from the
   apps-discuss@ietf.org mailing list (or a successor designated by the
   APPS Area Directors). Before a period of 30 days has passed, the
   Designated Expert will either approve or deny the registration
   request, communicating this decision both to the review list and to
   IANA. Denials should include an explanation and, if applicable,
   suggestions as to how to make the request successful.

   The Link Relation Type registry's initial contents are:

   o Relation Name: alternate
   o Description: Designates a substitute for the link's context.
   o Reference: [W3C.REC-html401-19991224]

   o Relation Name: appendix
   o Description: Refers to an appendix.
   o Reference: [W3C.REC-html401-19991224]

o Relation Name: bookmark
o Description: Refers to a bookmark or entry point.
o Reference: [W3C.REC-html401-19991224]

o Relation Name: chapter
o Description: Refers to a chapter in a collection of resources.
o Reference: [W3C.REC-html401-19991224]

o Relation Name: contents
o Description: Refers to a table of contents.
o Reference: [W3C.REC-html401-19991224]

o Relation Name: copyright
o Description: Refers to a copyright statement that applies to the
   link's context.
o Reference: [W3C.REC-html401-19991224]

o Relation Name: current
o Description: Refers to a resource containing the most recent
   item(s) in a collection of resources.
o Reference: [RFC5005]

o Relation Name: describedby
o Description: Refers to a resource providing information about the
   link's context.
o Documentation: <http://www.w3.org/TR/powder-dr/>

o Relation Name: edit
o Description: Refers to a resource that can be used to edit the
   link's context.
o Reference: [RFC5023]

o Relation Name: edit-media
o Description: Refers to a resource that can be used to edit media
   associated with the link's context.
o Reference: [RFC5023]

o Relation Name: enclosure
o Description: Identifies a related resource that is potentially
   large and might require special handling.
o Reference: [RFC4287]

o Relation Name: first
o Description: An IRI that refers to the furthest preceding resource
   in a series of resources.
o Reference: [this document]

o Relation Name: glossary
o Description: Refers to a glossary of terms.
o Reference: [W3C.REC-html401-19991224]

o Relation Name: help
o Description: Refers to a resource offering help (more information,
   links to other sources information, etc.)
o Reference: [W3C.REC-html401-19991224]

o Relation Name: index
o Description: Refers to an index.
o Reference: [W3C.REC-html401-19991224]

o Relation Name: last
o Description: An IRI that refers to the furthest following resource
   in a series of resources.
o Reference: [this document]

o Relation Name: license
o Description: Refers to a license associated with the link's
   context.
o Reference: [RFC4946]

o Relation Name: next
o Description: Refers to the next resource in a ordered series of
   resources.
o Reference: [W3C.REC-html401-19991224]

o Relation Name: next-archive
o Description: Refers to the immediately following archive resource.
o Reference: [RFC5005]

o Relation Name: payment
o Description: indicates a resource where payment is accepted.
o Reference: [this document]

o Relation Name: prev
o Description: Refers to the previous resource in an ordered series
   of resources. Synonym for "previous".
o Reference: [W3C.REC-html401-19991224]

o Relation Name: previous
o Description: Refers to the previous resource in an ordered series
   of resources. Synonym for "prev".
o Reference: [W3C.REC-html401-19991224]

o   Relation Name: prev-archive
o   Description: Refers to the immediately preceding archive resource.
o   Reference: [RFC5005]

o   Relation Name: related
o   Description: Identifies a related resource.
o   Reference: [RFC4287]

o   Relation Name: replies
o   Description: Identifies a resource that is a reply to the context
    of the link.
o   Reference: [RFC4685]

o   Relation Name: section
o   Description: Refers to a section in a collection of resources.
o   Reference: [W3C.REC-html401-19991224]

o   Relation Name: self
o   Description: Conveys an identifier for the link's context.
o   Reference: [RFC4287]

o   Relation Name: service
o   Description: Indicates a URI that can be used to retrieve a
    service document.
o   Reference: [RFC5023]
o   Notes: When used in an Atom document, this relation specifies Atom
    Publishing Protocol service documents by default.

o   Relation Name: start
o   Description: Refers to the first resource in a collection of
    resources.
o   Reference: [W3C.REC-html401-19991224]

o   Relation Name: stylesheet
o   Description: Refers to an external style sheet.
o   Reference: [W3C.REC-html401-19991224]

o   Relation Name: subsection
o   Description: Refers to a resource serving as a subsection in a
    collection of resources.
o   Reference: [W3C.REC-html401-19991224]

o   Relation Name: up
o   Description: Refers to a parent document in a hierarchy of
    documents.
o   Reference: [this document]

   o Relation Name: via
   o Description: Identifies a resource that is the source of the
      information in the link's context.
   o Reference: [RFC4287]

7. Security Considerations

   The content of the Link header-field is not secure, private or
   integrity-guaranteed, and due caution should be exercised when using
   it.

   Applications that take advantage of typed links should consider the
   attack vectors opened by automatically following, trusting, or
   otherwise using links gathered from HTTP headers. In particular,
   Link headers that use the "anchor" parameter to associate a link's
   context with another resource should be treated with due caution.

8. Internationalisation Considerations

   Target IRIs may need to be converted to URIs in order to express them
   in serialisations that do not support IRIs. This includes the Link
   HTTP header.

   Similarly, the anchor parameter of the Link header does not support
   IRIs, and therefore IRIs must be converted to URIs before inclusion
   there.

   Relation types are defined as URIs, not IRIs, to aid in their
   comparison. It is not expected that they will be displayed to end
   users.

9. References

9.1. Normative References

   [RFC2026] Bradner, S., "The Internet Standards Process -- Revision
              3", BCP 9, RFC 2026, October 1996.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119, March 1997.

   [RFC2231] Freed, N. and K. Moore, "MIME Parameter Value and Encoded
              Word Extensions: Character Sets, Languages, and
              Continuations", RFC 2231, November 1997.

   [RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H.,
              Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext
              Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999.

   [RFC3864] Klyne, G., Nottingham, M., and J. Mogul, "Registration
              Procedures for Message Header Fields", BCP 90, RFC 3864,
              September 2004.

   [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform
              Resource Identifier (URI): Generic Syntax", STD 66,
              RFC 3986, January 2005.

   [RFC3987] Duerst, M. and M. Suignard, "Internationalized Resource
              Identifiers (IRIs)", RFC 3987, January 2005.

   [RFC4288] Freed, N. and J. Klensin, "Media Type Specifications and
              Registration Procedures", BCP 13, RFC 4288, December 2005.

   [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an
              IANA Considerations Section in RFCs", BCP 26, RFC 5226,
              May 2008.

9.2. Informative References

   [RFC2068] Fielding, R., Gettys, J., Mogul, J., Nielsen, H., and T.
              Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1",
              RFC 2068, January 1997.

   [RFC4287] Nottingham, M. and R. Sayre, "The Atom Syndication
              Format", RFC 4287, December 2005.

   [RFC4685] Snell, J., "Atom Threading Extensions", RFC 4685,
              September 2006.

   [RFC4946] Snell, J., "Atom License Extension", RFC 4946, July 2007.

   [RFC5005] Nottingham, M., "Feed Paging and Archiving", RFC 5005,
              September 2007.

   [RFC5023] Gregorio, J. and B. de hOra, "The Atom Publishing
              Protocol", RFC 5023, October 2007.

   [RFC5137] Klensin, J., "ASCII Escaping of Unicode Characters",
              BCP 137, RFC 5137, February 2008.

   [W3C.REC-html401-19991224]
              Raggett, D., Hors, A., and I. Jacobs, "HTML 4.01
              Specification", W3C REC REC-html401-19991224,

                December 1999.

    [W3C.REC-rdfa-syntax-20081014]
                Pemberton, S., Birbeck, M., Adida, B., and S. McCarron,
                "RDFa in XHTML: Syntax and Processing", World Wide Web
                Consortium Recommendation REC-rdfa-syntax-20081014,
                October 2008,
                <http://www.w3.org/TR/2008/REC-rdfa-syntax-20081014>.

    [W3C.REC-xhtml-basic-20080729]
                Baker, M., Wugofski, T., Ishikawa, M., Stark, P., Matsui,
                S., and T. Yamakami, "XHTML[TM] Basic 1.1", World Wide Web
                Consortium Recommendation REC-xhtml-basic-20080729,
                July 2008,
                <http://www.w3.org/TR/2008/REC-xhtml-basic-20080729>.

Appendix A. Notes on Using the Link Header with HTML4

HTML motivated the original syntax of the Link header, and many of
the design decisions in this document are driven by a desire to stay
compatible with these uses.

In HTML4, the link element can be mapped to links as specified here
by using the "href" attribute for the target URI, and "rel" to convey
the relation type, as in the Link header. The context of the link is
the URI associated with the entire HTML document.

HTML4 also has a "rev" parameter for links that allows a link's
relation to be reversed. The Link header has a "rev" parameter to
allow the expression of these links in HTTP headers, but its use is
not encouraged, due to the confusion this mechanism causes as well as
conflicting interpretations among HTML versions.

All of the link relations defined by HTML4 have been included in the
link relation registry, so they can be used without modification.
However, extension link relations work differently in HTML4 and the
Link header; the former uses a document-wide "profile" URI to scope
the relations, while the latter allows the use of full URIs on
individual relations.

Therefore, when using the profile mechanism in HTML4, it is necessary
to map the profiled link relations to URIs when expressed in Link
headers. For example, in HTML:

```
<html>
  <head profile="http://example.com/profile1/">
    <link rel="foo" href="/bar">
  </head>
  [...]
```

could be represented as a header like this;

Link: </bar>; rel="http://example.com/profile1/foo"

Profile authors should note this when creating profile URIs; it may
be desirable to use URIs that end in a delimiter (e.g., "/" or "#"),
to make extracting the specific relation in use easier.

Note that RDFa [W3C.REC-rdfa-syntax-20081014] defines a different way
to map link relations to URIs in XHTML
[W3C.REC-xhtml-basic-20080729]. Although this convention is not
defined for HTML4, some authors may still use it there.

Surveys of existing HTML content have shown that unregistered link
relation types that are not URIs are (perhaps inevitably) common.
Consuming HTML implementations should not consider such unregistered
short links to be errors, but rather relation types with a local
scope (i.e., their meaning is specific and perhaps private to that
document).

HTML4 also defines several attributes on links that are not
explicitly defined by the Link header. These attributes can be
serialised as link-extensions to maintain fidelity.

Finally, the HTML4 specification gives a special meaning when the
"alternate" and "stylesheet" relations coincide in the same link.
Such links should be serialised in the Link header using a single
list of relation-types (e.g., rel="alternate stylesheet") to preserve
this relationship.

Appendix B. Notes on Using the Link Header with Atom

Atom conveys links in the atom:link element, with the "href"
attribute indicating the target IRI and the "rel" attribute
containing the relation type. The context of the link is either a
feed IRI or an entry ID, depending on where it appears; generally,
feed-level links are obvious candidates for transmission as a Link
header.

When serialising an atom:link into a Link header, it is necessary to

convert target IRIs (if used) to URIs.

Atom defines extension relation types in terms of IRIs. This
specification re-defines them as URIs, to simplify and reduce errors
in their comparison.

Atom allows registered link relation types to be serialised as
absolute URIs. Such relation types SHOULD be converted to the
appropriate registered form (e.g.,
"http://www.iana.org/assignments/relation/self" to "self") so that
they are not mistaken for extension relation types.

Furthermore, Atom link relations are always compared in a case-
sensitive fashion; therefore, registered link relations SHOULD be
converted to their registered form (usually, lower case) when
serialised in an Atom document.

Note also that while the Link header allows multiple relations to be
serialised in a single link, atom:link does not. In this case, a
single link-value may map to several atom:link elements.

As with HTML, atom:link defines some attributes that are not
explicitly mirrored in the Link header syntax, but they may also be
used as link-extensions to maintain fidelity.

Appendix C. Defining New Link Serialisations

New serialisations of links (as defined by this specification) need
to address several issues, including:

o Specific syntax for each component of the link model described in
   Section 3.
o What target attributes, if any, are defined by the serialisation.
o How to determine the context of the link.
o How to differentiate registered link relations from extension link
   relations (if the latter are serialised as URIs, this is
   relatively straightforward).

Appendix D. Acknowledgements

This specification lifts the idea and definition for the Link header
from RFC2068; credit for it belongs entirely to the authors of and
contributors to that document. The link relation registrations
themselves are sourced from several documents; see the applicable
references.

Appendix E. Document history

[[to be removed by the RFC editor before publication as an RFC.]]

-06

o Added "up" and "service" relation types.
o Fixed "type" attribute syntax and added prose.
o Added note about RDFa and XHTML to HTML4 notes.
o Removed specific location for the registry, since IANA seems to
   have its own ideas about that.

-05

o Clarified how to resolve relative URIs in the 'anchor' parameter.
o Tweaked language about dereferencing relation type URIs.
o Separated out examples.
o Made target-parameters more explicit in the model.
o Discourage special semantics between different relations, or based
   upon cardinality.
o Grandfathered in special semantics of 'alternate stylesheet' for
   HTML4.
o Note that extension types can be serialised in ways other than as
   URIs, as long as they can be converted to URIs.
o Change default context of a link header to that of the requested
   resource.
o Use this document as reference for relations that don't have a
   formal definition other than the registry entries; avoids circular
   references.
o Noted that ordering of links is not significant or defined in this
   spec, but may be in specific applications.
o Adjusted uses of 'application' to 'serialisation' where
   appropriate.
o Added 'Defining New Link Serialisations' section.
o Added note about case sensitivity when comparing registered
   relation types in Atom.

-04

o Defined context as a resource, rather than a representation.
o Removed concept of link directionality; relegated to a deprecated
   Link header extension.

o Relation types split into registered (non-URI) and extension
   (URI).
o Changed wording around finding URIs for registered relation types.
o Changed target and context URIs to IRIs (but not extension
   relation types).
o Add RFC2231 encoding for title parameter, explicit BNF for title*.
o Add i18n considerations.
o Specify how to compare relation types.
o Changed registration procedure to Designated Expert.
o Softened language around presence of relations in the registry.
o Added describedby relation.
o Re-added 'anchor' parameter, along with security consideration for
   third-party anchors.
o Softened language around HTML4 attributes that aren't directly
   accommodated.
o Various tweaks to abstract, introduction and examples.

-03

o Inverted focus from Link headers to link relations.
o Specified was a link relation type is.
o Based on discussion, re-added 'rev'.
o Changed IESG Approval to IETF Consensus for relation registrations
   (i.e., require a document).
o Updated RFC2434 reference to RFC5226.
o Registered relations SHOULD conform to sgml-name.
o Cautioned against confusing relation types with media types.

-02

o Dropped XLink language.
o Removed 'made' example.
o Removed 'rev'. Can still be used as an extension.
o Added HTML reference to introduction.
o Required relationship values that have a ; or , to be quoted.
o Changed base URI for relation values.
o Noted registry location.
o Added advisory text about HTML profile URIs.
o Disallowed registration of relations that only differ in case.
o Clarified language about IRIs in Atom.
o Added descriptions for 'first', 'last', and 'payment', referring
   to current IANA registry entries, as these were sourced from
   e-mail. Will this cause self-referential implosion?
o Explicitly updates RFC4287.
o Added 'type' parameter.
o Removed unnecessary advice about non-HTML relations in HTML
   section.

   -01

   o Changed syntax of link-relation to one or more URI; dropped
      Profile.
   o Dropped anchor parameter; can still be an extension.
   o Removed Link-Template header; can be specified by templates spec
      or elsewhere.
   o Straw-man for link relation registry.

   -00

   o Initial draft; normative text lifted from RFC2068.

Author's Address

   Mark Nottingham

   Email: mnot@mnot.net
   URI: http://www.mnot.net/