Network Working Group E. Wilde
Internet-Draft CA Technologies
Intended status: Informational H. Van de Sompel
Expires: February 3, 2018 Los Alamos National Laboratory
August 2, 2017

Linkset: A Link Relation Type and Media Types for Link Sets
draft-wilde-linkset-00

Abstract

This specification defines a link relation type and media types for working with sets of links. Using this link relation type and/or the media types is useful when it becomes necessary to represent links outside the context of a resource they are linking. One typical example are scenarios in which the number of links to put in an HTTP Link header field is too big, and thus these links should become a resource of their own.

Note to Readers

Please discuss this draft on the ART mailing list (<https://www.ietf.org/mailman/listinfo/art>).

Online access to all versions and files is available on GitHub (<https://github.com/dret/I-D/tree/master/linkset>).

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Resources on the Web often convey typed Web Links [I-D.nottingham-rfc5988bis] as a part of resource representations, for example, using the <link> element for HTML representations, or the "Link" header field in HTTP response headers for representations of any media type. In some cases, however, providing links by value is impractical or impossible. In these cases, an approach to provide links by reference (instead of by value) can solve the problem. This specification defines the "linkset" relation type that allows to link resources to sets of links, thereby making it possible to represent links by reference, and not by value.

2. Terminology

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 RFC 2119 [RFC2119].

This section introduces two types of resources involved in providing links by reference, as well as the link relation type used to interlink them.

2.1. Origin Resource

An "origin resource" is a resource that makes links in which it participates discoverable by providing a typed link that has a "link set resource" as the target. From the perspective of the origin resource, the links in the "link set resource" are therefore provided by reference.

2.2. Link Set Resource

A "link set resource" is a resource - distinct from the origin resource, and possibly linked to from the origin resource - that provides one or more links in which the origin resource participates.

Because the link set resource is distinct from the origin resource, links provided by the link set resource must represent both the source and target of each link to allow unambiguous interpretation.

Link set resources can be represented in any way that allows representation of links in a way that supports both source and target anchors of links. Section Section 5 defines two possible representations, both of which are based on the link model defined by Web Linking [I-D.nottingham-rfc5988bis].

3. Scenarios

The following sections outline some scenarios in which it is useful to have the ability to separate resources and links pertaining to them.

These are all scenarios in which providing (some) links by reference is advantageous or necessary to accomplish certain goals. It is important to keep in mind that even when using the pattern of "links by reference", it is still possible to also provide links by value, allowing resources to decide and combine which of the two patterns they would like to use.

3.1. Third-Party Links

In some cases, it is useful that links pertaining to an origin resource are provided by a server other than the one that hosts the origin resource. For example, this allows:

- Providing links in which the origin resource is involved not just as source but also as target.

- Providing links pertaining to the original resource that the server hosting that resource is not aware of.

- External management of links pertaining to the origin resource in a special-purpose link management service.

In such cases, a third-party link set resource provides links pertaining to the origin resource. This link set resource may be managed by the same custodian as the origin resource, or by a third party.

In order for the server hosting the origin resource to provide an up-to-date and complete set of links for it, it would need to obtain the links from the link set resource, and embed them in the origin resource's representations prior to responding to a client. Doing so would increase latency and load, which may be unnecessary if a client is not intent on consuming these links. Providing links by reference, instead of by value, removes the server-to-server communication and resulting overhead required to obtain links. Instead, the consumer of the origin resource can decide if they need the additional links as context for the resource.

3.2. Challenges Writing to HTTP Header Field

In some cases, it is not straightforward to write links to the HTTP Link header from an application. This can, for example, be the case because not all required link information is available to the application or because the application does not have the capability to directly write HTTP headers. In such cases, providing links by reference can be a solution because a link to a link set pertaining to a resource can typcially be added by means of a web server rewrite rule that leverages the resource's URI.

3.3. Large Number of Links

When conveying links in the HTTP "Link" header, it is possible for the size of the HTTP response header to become unpredictable. This can be the case when links are determined dynamically dependent on a range of contextual factors. It is possible to statically configure a web server to correctly handle large HTTP response headers by specifying an upper boundary for their size. But when the number of links is unpredictable, estimating a reliable upper boundary is challenging.

HTTP [RFC7231] defines error codes related to excess communication by the user agent ("413 Request Entity Too Large" and "414 Request-URI Too Long"), but no specific error codes are defined to indicate that response header content exceeds the upper boundary that can be handled by the server, and thus it has been truncated. As a result, applications take counter measures aimed at controlling the size of the HTTP "Link" header, for example by limiting the links they provide to those with select relation types, thereby limiting the value of the HTTP "Link" header to clients. Providing links by reference, instead of by value, overcomes challenges related to the unpredictable nature of the extent of HTTP "Link" headers.

In more extreme scenarios it is conceivable that the number of links pertaining to the origin resource becomes so large that the response from the associated link set resource becomes too large. This could be the case for highly popular origin resources, when the link set includes incoming links as well. In such cases, the link set resource could deliver responses incrementally, for example, using a paged resource model that clients can consume as required, requesting links incrementally by paging through the provided link set.

4. The "linkset" Relation Type for Linking to Link Sets

A link with the "linkset" link relation type has as Context IRI the IRI of an origin resource, and as Target IRI the IRI of an associated link set resource.

A link with the "linkset" relation type MAY be provided in the header and/or the body of the origin resource's representation. It may also be discovered by other means, such as through client-side information.

More than one link with a "linkset" relation type MAY be provided. Multiple such links can refer to the same set of links expressed using different representations, or to different link sets (potentially provided by different services).

The use of a link with the "linkset" relation type does not preclude the provision of links with other relation types, i.e. the origin resource can provide typed links other than a "linkset" link. Therefore, the effective set of links pertaining to the origin resource is the union of the links that the resource itself provides, and of all links in which it participates which are provided by the link set resources linked from it via "linkset" links.

The link set resource MAY provide the links that pertain to the origin resource in its HTTP response header and/or body:

o In cases where the link set resource provides these links in its Link HTTP response header, the payload of that header MUST comply with the syntax defined in Web Linking [I-D.nottingham-rfc5988bis]. The media type of the response body is not constrained.

o In cases where the link set resource provides these links in its response body, the body SHOULD allow a client to determine the source and target of each provided link. The media type of the response body is otherwise not constrained.

There is no constraint on the Target IRI of a link with the "linkset" relation type; designing and using these links is left to the discretion of implementers.

If an origin resource provides a "linkset" link pointing at a link set resource, and that link set resource provides a "linkset" link in turn, then this latter link points at links pertaining to the link set resource. This means that in the context of the latter link, the link set resource is an origin resource. This means that linkset relations are not transitive, and it is up to a client to decide whether they follow "nested chains" of linkset links or not.

5. Document Formats for Link Sets

This section specifies two document formats to convey link sets, one that is identical to the payload of the HTTP Link header as specified in RFC 5988bis [I-D.nottingham-rfc5988bis], and the other is a JSON-based format that does not have the character encoding limitations to which HTTP headers are subject as per RFC 5987 [RFC5987].

5.1. Native Document Format for Link Sets: application/linkset

This document format is identical to the payload of the HTTP Link header. It is defined in Section 3 of RFC 5988bis [I-D.nottingham-rfc5988bis], more specifically by its ABNF production rule for "Link" and subsequent ones. The assigned media type for this format is application/linkset.

5.2. JSON Document Format for Link Sets: application/linkset+json

For applications that prefer a JSON serialization of link set resources, the following definition provides a JSON [RFC7159] serialization which is intended to faithfully reproduce the abstract model of RFC 5988bis [I-D.nottingham-rfc5988bis].

In the JSON representation, a link set resource is represented by an array, where each member of the array is an object, which is the JSON representation of a link value. Implementations MUST always wrap link value representations in an array, even if a link set contains only one link value. Implementations MUST NOT include any members other than link value objects in the array representing a link set.

5.2.1. Link Value Objects

A link value object represents an individual link of a link set. Each link value object is represented as a JSON object and MUST have an "href" member that represents the link target. The value of the "href" member is a URI-Reference that conveys the target IRI.

```
[ { "href" : "http://example.com/foo" } ]
```

This minimal example of a JSON-serialized link set contains just one link which only uses the link target URI in its representation. In most cases, links are further qualified by link parameters, which are serialized as additional members of the link value object.

5.2.2. Link Parameters

In accordance with RFC 5988bis, link parameters apply to individual links, and while a number of link parameters are defined by RFC 5988bis, other link parameters MAY occur and implementations MUST ignore the ones that they do not understand.

Link parameters appear as members of link value objects. Their name is the name of the link parameter, and their value is the value of the link parameter.

```
[ { "href" : "http://example.com/foo",
    "rel"  :  "next" } ]
```

This example add a link relation type to the link, using the "rel" link parameter defined by RFC 5988bis.

RFC 5988bis defines the link parameters "rel", "anchor", "rev", "hreflang", "media", "type", "title", and "title*". For the link parameters "rel", "anchor", "rev", "hreflang", "media", and "type", these can appear in the JSON serialization in the same way as shown in the above example, as members of link value objects that have the link parameter type as their name, and its value as their value. The case for "title" and "title*" is different because these have been specifically introduced to deal with character encoding issues in HTTP link headers, and can be encoded more effectively in JSON.

5.2.3. Internationalized Link Parameters

While most link parameters can be translated directly from their RFC 5988bis variants, one special case are link parameters following the model of RFC 5987 [RFC5987]. These link parameters use the pattern of the "title" and "title*" link parameters defined by RFC 5988bis. The model defined by RFC 5987 [RFC5987] and used by "title" and "title*" may be used by additional link parameters as well, which can be recognized by the naming scheme of using a link parameter name and its variant with a trailing asterisk. The JSON serialization model described here applies to all these parameters and MUST be used for the JSON representation of link sets. In other words, implementations MUST NOT directly translate RFC 5987 style parameters into corresponding JSON link value object members. Instead they MUST map them to the JSON representation defined in this section.

Internationalized link parameters use the link parameter name as their name, and their value is either a string representing the link parameter value, or an object representing one or more language tagged link parameter values. In such an object, the set of members uses a language tag [RFC5646] as their names, and their values are

strings representing the link parameter values associated with the respective language tag.

6. Examples

Sections Section 6.1 and Section 6.2 show examples whereby the link set resource provides links pertaining to the origin resource, in its response header and body, respectively.

6.1. Links Provided in the Header of the Link Set Resource

Figure 1 shows a client issuing an HTTP head request against origin resource http://example.org/resource1.

```
   HEAD /resource1 HTTP/1.1
   Host: example.org
   Connection: close
```

Figure 1: Client HTTP HEAD Request

Figure 2 shows the response to the HEAD request of Figure 1. The response contains a Link header with a link that uses the "linkset" relation type. It indicates that links pertaining to the origin resource are provided by link set resource http://example.com/links?uri=http%3A%2F%2Fexample.org%2Fresource.

```
HTTP/1.1 200 OK
Date: Mon, 28 Nov 2016 14:37:51 GMT
Server: Apache-Coyote/1.1
Link: <http://example.com/links?uri=http%3A%2F%2Fexample.org%2Fresource>
      ; rel="linkset"
      ; type="text/html"
Content-Length: 5214
Content-Type: text/html;charset=utf-8
Connection: close
```

Figure 2: Response to HTTP HEAD on Origin Resource

While in this example the IRI of the linkset resource uses a pattern that represents the IRI of the origin resource, this is opaque to the client, which simply follows the provided linkset IRI when retrieving the linkset resource.

Figure 3 shows the client issuing an HTTP GET request against the link set resource provided in Figure 2.

```
GET /links?uri=http%3A%2F%2Fexample.org%2Fresource HTTP/1.1
Host: example.com
Connection: close
```

Figure 3: Client HTTP GET against the Link Set Resource

Figure 4 shows the response headers to the HTTP GET request of Figure 3. The links pertaining to the origin resource are provided in the Link response header of the link set resource. As can be seen, in order to support an unambiguous determination of the Context IRI of each link, the "anchor" attribute is provided for each link. Note that most, but not all, links have the origin resource as Context IRI (anchor).

```
HTTP/1.1 200 OK
Date: Mon, 28 Nov 2016 14:40:02 GMT
Server: Apache-Coyote/1.1
Link: <http://authors.example.net/johndoe>
   ; rel="author"
   ; type="application/rdf+xml"
   ; anchor="http://example.org/resource1",
 <http://authors.example.net/janedoe>
   ; rel="author"
   ; type="application/rdf+xml"
   ; anchor="http://example.org/resource1",
 <http://example.org/resource1/items/AF48EF.pdf>
   ; rel="item"
   ; type="application/pdf"
   ; anchor="http://example.org/resource1",
 <http://example.org/resource1/items/CB63DA.html>
   ; rel="item"
   ; type="text/html"
   ; anchor="http://example.org/resource1",
 <http://example.net/resource41/>
   ; rel="related"
   ; type="application/pdf"
   ; anchor="http://example.org/resource1/items/AF48EF.pdf"
Content-Type: text/html
Content-Length: 3018
```

Figure 4: Response to HTTP GET against the Link Set Resource

6.2. Links Provided in the Body of the Link Set Resource, Link Set Serialized as application/linkset+json

Figure 5 is an example of a client issuing an HTTP head request against origin resource http://example.org/article?id=10.1371/journal.pone.0167475

```
HEAD article?id=10.1371/journal.pone.0167475 HTTP/1.1
Host: example.org
Connection: close
```

Figure 5: Client HTTP HEAD Request

Figure 6 shows the response to the HEAD request of Figure 5. The response contains a Link header with a link that has the "linkset" relation type. It indicates that links pertaining to the origin resource are provided by link set resource http://example.com/links/10.1371/journal.pone.0167475, which provides a representation with vendor media type application/vnd.example.org.linkset+json.

```
HTTP/1.1 200 OK
Date: Mon, 28 Nov 2016 14:37:51 GMT
Server: Apache-Coyote/1.1
Link: <http://example.com/links/10.1371/journal.pone.0167475>
      ; rel="linkset"
      ; type="application/linkset+json"
Content-Length: 236
Content-Type: text/html;charset=utf-8
Connection: close
```

Figure 6: Response to HTTP HEAD on Origin Resource

In this example, the IRI of the linkset resource does not directly represent the IRI of the origin resource anymore. There still is an association possible through a IRI pattern that is including DOI information, but as in the example above, the linkset IRI is opaque to the client which simply accesses the IRI to retrieve the linkset resource.

Figure 7 shows the client issuing an HTTP GET request against the link set resource provided in Figure 6.

```
   GET /links/10.1371/journal.pone.0167475 HTTP/1.1
   Host: example.com
   Accept: application/linkset+json
   Connection: close
```

Figure 7: Client HTTP GET against the Link Set Resource

Figure 8 shows the response headers to the HTTP GET request of Figure 7. The links pertaining to the origin resource are provided in the response body of the link set resource and are serialized according to the vendor media type application/vnd.example.org.linkset+json.

```
HTTP/1.1 200 OK
Date: Mon, 28 Nov 2016 14:40:02 GMT
Server: Apache-Coyote/1.1
Content-Type: application/linkset+json
Content-Length: 794

[{"href":"http://authors.example.net/johndoe","anchor":"http://example.org/
article?id=10.1371/journal.pone.0167475","rel":"author","type":"application/
rdf+xml"},
 {"href":"http://authors.example.net/janedoe","anchor":"http://example.org/
article?id=10.1371/journal.pone.0167475","rel":"author","type":"application/
rdf+xml"},
 {"href":"http://example.org/resource1/items/AF48EF.pdf","anchor":"http://
example.org/article?id=10.1371/journal.pone.0167475","rel":"item","type":"text/
html"},
 {"href":"http://example.org/resource1/items/CB63DA.html","anchor":"http://
example.org/article?id=10.1371/journal.pone.
0167475","rel":"item","type":"application/pdf"},
 {"href":"http://example.net/resource41/","anchor":"http://example.org/
resource1/items/AF48EF.pdf","rel":"related","type":"application/pdf"}]
```

Figure 8: Response to HTTP GET against the Link Set Resource

If Figure 6 would have provided a link to a link set with media type application/linkset, and if the client would have requested that link set, then the body of the response would have been similar to Figure 8. But it would have had application/linkset as Content-Type, the payload of the HTTP Link header of Figure 4 as body, and an accordingly adjusted value for Content-Length.

7. IANA Considerations

7.1. Link Relation Type: linkset

The link relation type below has been registered by IANA per Section 6.2.1 of Web Linking [I-D.nottingham-rfc5988bis]:

Relation Name: linkset

Description: The Target IRI of a link with the "linkset" relation type provides a set of links that pertain to the Context IRI of the link.

Reference: [[This document]]

7.2. Media Type: application/linkset

7.2.1. IANA Considerations

The Internet media type [RFC6838] for a natively encoded link set is application/linkset.

Type name: application

Subtype name: linkset

Required parameters: none

Optional parameters: none

Encoding considerations: ...

Security considerations: ...

Interoperability considerations: ...

Published specification: [[This document]]

Applications that use this media type: ...

Additional information:

Magic number(s): N/A

File extension(s): This media type does not propose a specific extension.

Macintosh file type code(s): TEXT

Person & email address to contact for further information: Herbert Van de Sompel <herbertv@lanl.gov>

Intended usage: COMMON

Restrictions on usage: none

Author: Herbert Van de Sompel <herbertv@lanl.gov>

Change controller: IETF

7.3. Media Type: application/linkset+json

The Internet media type [RFC6838] for a JSON-encoded link set is application/linkset+json.

Type name: application

Subtype name: linkset+json

Required parameters: none

Optional parameters: none

Encoding considerations: ...

Security considerations: ...

Interoperability considerations: ...

Published specification: [[This document]]

Applications that use this media type: ...

Additional information:

Magic number(s): N/A

File extension(s): JSON documents often use ".json" as the file extension, and this media type does not propose a specific extension other than this generic one.

Macintosh file type code(s): TEXT

Person & email address to contact for further information: Herbert Van de Sompel <herbertv@lanl.gov>

Intended usage: COMMON

Restrictions on usage: none

Author: Herbert Van de Sompel <herbertv@lanl.gov>

Change controller: IETF

8. Security Considerations

...

9. Normative References

[I-D.nottingham-rfc5988bis]
Nottingham, M., "Web Linking", draft-nottingham-rfc5988bis-06 (work in progress), June 2017.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <http://www.rfc-editor.org/info/rfc3986>.

[RFC5646] Phillips, A., Ed. and M. Davis, Ed., "Tags for Identifying Languages", BCP 47, RFC 5646, DOI 10.17487/RFC5646, September 2009, <http://www.rfc-editor.org/info/rfc5646>.

[RFC5987] Reschke, J., "Character Set and Language Encoding for Hypertext Transfer Protocol (HTTP) Header Field Parameters", RFC 5987, DOI 10.17487/RFC5987, August 2010, <http://www.rfc-editor.org/info/rfc5987>.

[RFC6690] Shelby, Z., "Constrained RESTful Environments (CoRE) Link Format", RFC 6690, DOI 10.17487/RFC6690, August 2012, <http://www.rfc-editor.org/info/rfc6690>.

[RFC6838] Freed, N., Klensin, J., and T. Hansen, "Media Type Specifications and Registration Procedures", BCP 13, RFC 6838, DOI 10.17487/RFC6838, January 2013, <http://www.rfc-editor.org/info/rfc6838>.

[RFC7159] Bray, T., Ed., "The JavaScript Object Notation (JSON) Data Interchange Format", RFC 7159, DOI 10.17487/RFC7159, March 2014, <http://www.rfc-editor.org/info/rfc7159>.

[RFC7231] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content", RFC 7231, DOI 10.17487/RFC7231, June 2014, <http://www.rfc-editor.org/info/rfc7231>.

[W3C.REC-html401-19991224]
Rivoal, F., "Media Queries", World Wide Web Consortium Recommendation REC-css3-mediaqueries-20120619, June 2012.

Authors' Addresses

Erik Wilde
CA Technologies

Email: erik.wilde@dret.net
URI: http://dret.net/netdret/

Herbert Van de Sompel
Los Alamos National Laboratory

Email: herbertv@lanl.gov
URI: http://public.lanl.gov/herbertv/