

The Deprecation HTTP Header
draft-dalal-deprecation-header-00

Abstract

The HTTP Deprecation response header can be used to signal to consumers of a URI-identified resource that the use of the resource has been deprecated. Additionally, the deprecation link relation can be used to link to a resource that provides additional context for the deprecation, and possibly ways in which clients can find a replacement for the deprecated resource.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on August 30, 2019.

Copyright Notice

Copyright (c) 2019 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 (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in [Section 4](#).e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	2
1.1.	Notational Conventions	3
2.	The Deprecation HTTP Response Header Field	3
2.1.	Syntax	3
2.1.1.	Version	4
2.1.2.	Date	4
3.	The Deprecation Link Relation Type	5
3.1.	Documentation	5
3.2.	Policy	6
4.	Recommend Replacement	6
5.	Sunset	6
6.	IANA Considerations	7
6.1.	The Deprecation Response Header Field	7
6.2.	The Deprecation Link Relation Type	7
7.	Implementation Status	8
8.	Security Considerations	9
9.	Example	10
10.	References	10
10.1.	Normative References	10
10.2.	Informative References	11
Appendix A.	Acknowledgments	11
	Authors' Addresses	12

[1.](#) Introduction

Deprecation of a URI-identified resource is a technique to communicate information about the lifecycle of a resource. It encourages applications to migrate away from the resource and discourage applications from forming new dependencies on the resource, and informs applications of the risk of continuing dependence upon the resource.

The act of deprecation does not change any behavior of the resource. It just informs client of the fact that a resource is deprecated. The Deprecation HTTP response header field MAY be used to convey this fact at runtime to clients. The header field can carry additional information such as since when the deprecation is in effect.

In addition to the Deprecation header field the resource provider can use other header fields to convey additional information related to deprecation. For example, information such as where to find documentation related to the deprecation or what should be used as an alternate and when the deprecated resource would be unreachable, etc.

Alternates of a resource can be similar resource(s) or a newer version of the same resource.

1.1. Notational Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [BCP 14](#) [[RFC2119](#)] [[RFC8174](#)] when, and only when, they appear in all capitals, as shown here.

This specification uses the Augmented Backus-Naur Form (ABNF) notation of [[RFC5234](#)] and includes, by reference, the "token" rule, DQUOTE (double quote) rule, the SP (space) rule and the "rule" extension that allows for compact definition of comma-separated lists using a '#' operator (similar to how the '*' operator indicates repetition), HTTP-date rule as defined within Sections [3.2.6](#) and [7](#) of [[RFC7230](#)] and [Section 7.1.1 of](#) [[RFC7231](#)].

2. The Deprecation HTTP Response Header Field

The "Deprecation" HTTP response header field allows a server to communicate to a client that the URI-identified resource in context of the message is deprecated. It can also provide information that the resource is deprecated since which version.

2.1. Syntax

The "Deprecation" response header contains the header name "Deprecation" followed by a ":" and a property(s). Each property consists of a name-value-pair. Servers SHOULD NOT send Deprecation headers that fail to conform to the following grammar:

```
deprecation-header = "Deprecation:" SP "version"=vval, "date"=dval,  
                      *( extension )  
extension = property-name "=" property-value  
property-name = DQUOTE token DQUOTE  
token = <token, defined as in [RFC7230], Section 3.2.6>  
vval = property-value  
property-value = DQUOTE *( pchar ) DQUOTE  
pchar = %x23 / %x2B-3A / %x41-5A / %x61-7A / %x7C  
        ; US-ASCII characters  
dval = DQUOTE HTTP-date DQUOTE
```

Note that some of the grammatical terms above reference documents that use different grammatical notations than this document (which uses ABNF from [[RFC5234](#)]).

Servers SHOULD NOT include more than one "Deprecation" header field in the same response. If a server sends multiple responses containing "Deprecation" headers concurrently to the user agent (e.g., when communicating with the user agent over multiple sockets), these responses create a "race condition" that can lead to unpredictable behavior.

The value of "Deprecation" response header field could consist of at least 1 standard property: "date" or "version" as shown below. Either of "version" or "date" is REQUIRED and both are also allowed.

Deprecation: version="version", date="date"

2.1.1. Version

The value of the "version" property, if present, could be the version of the resource that is deprecated. The value of "version" would be an opaque version identifier. For resources that use date-based versioning scheme, the value would be accordingly.

Following example indicates that the version v1 of the resource in context is deprecated.

Deprecation: version="v1"

Following example shows that the version 2018-11-08 (November 8, 2018) of the resource in context is deprecated. Here the versioning scheme used is date-based.

Deprecation: version="2018-11-08"

2.1.2. Date

The value of "date" property, if present, would be the date when resource was deprecated. It would be in the form of a quoted HTTP-date timestamp, as defined in [Section 7.1.1.1 of \[RFC7231\]](#).

Following example shows that the resource in context is deprecated on Friday, November 11, 2018 at 23:59:59 GMT.

Deprecation: date="Fri, 11 Nov 2018 23:59:59 GMT"

Date could be in future too. If the value of "date" is in future, it means that the resource would be deprecated on the given date in future.

3. The Deprecation Link Relation Type

In addition to the Deprecation HTTP header, the server could use a "Link" header(s) to communicate to the client where to find more information about deprecation of the resource in context. This information could be in the form of documentation of the resource including details about the deprecation related aspects of the resource or the deprecation policy of the resource provider or both for example.

3.1. Documentation

For a URI-identified resource, deprecation could involve one or more parts of request, response or both. These parts could be one or more of the following.

- o URI - deprecation of one ore more query parameter(s) or path element(s)
- o method - HTTP method for the resource is deprecated
- o request header - one or more HTTP request header(s) is deprecated
- o response header - HTTP response header(s) is deprecated
- o request body - request body contains one or more deprecated element(s)
- o response body - response body contains one or more deprecated element(s)

The purpose of the "Deprecation" header is to provide just enough "hints" about the deprecation to the client application developer. It is safe to assume that on reception of the "Deprecation" header, the client developer would look up the resource's documentation in order to find deprecation related semantics. The resource developer could provide a link to the resource documentation using a "Link" header with relation type "deprecation" as shown below.

```
Deprecation: version="v1"  
Link: <https://developer.example.com/v1/customers>; rel="deprecation"  
      type="text/html"
```

where content at "https://developer.example.com/v1/customers" would be annotated to show deprecation of the relevant parts of the "customers" resource.

3.2. Policy

Resource provider would typically document versioning and deprecation policy with the resource documentation. To inform the client application developer of the deprecation policy, the resource provider could use the "deprecation" relation type as shown below.

```
Deprecation: version="v1"
```

```
Link: <https://developer.example.com/deprecation>; rel="deprecation"  
      type="text/html"
```

where deprecation policy of the resource provider "example.com" is described at "https://developer.example.com/deprecation".

4. Recommend Replacement

"Link" [[RFC8288](#)] header could be used in addition to the "Deprecation" header to recommend the client application about available alternates to the deprecated resource. Following relation types as defined in [[RFC8288](#)] are RECOMMENDED to use for the purpose.

- o "successor-version": Points to a resource containing the successor version. [[RFC5829](#)]
- o "latest-version": Points to a resource containing the latest (e.g., current) version. [[RFC5829](#)]
- o "alternate": Designates a substitute. [W3C.REC-html401-19991224]

Following example provides link to the successor version of the v1 version of "customer" resource that is deprecated.

```
Deprecation: version="v1"
```

```
Link: <https://api.example.com/v2/customers>; rel="successor-version"
```

This example provides link to an alternate resource to the "customer" resource that is deprecated.

```
Deprecation: version="2018-11-11"
```

```
Link: <https://api.example.com/v1/clients>; rel="alternate"
```

5. Sunset

In addition to the deprecation related information, if the resource provider wants to convey to the client application that the deprecated resource is expected to become unresponsive at a specific point in time, the [[Sunset](#)] header could be used in addition to the "Deprecation" header.

Following example indicates that the resource in context has been deprecated since version v2 and its sunset date is Friday, November 11, 2020 at 23:59:59 GMT.

```
Deprecation: version="v2"  
Sunset: Fri, 11 Nov 2020 23:59:59 GMT
```

Following example shows that the resource in context has been deprecated since Friday, November 11, 2018 at 23:59:59 GMT and its sunset date is Friday, November 11, 2020 at 23:59:59 GMT.

```
Deprecation: date="Fri, 11 Nov 2018 23:59:59 GMT"  
Sunset: Fri, 11 Nov 2020 23:59:59 GMT
```

6. IANA Considerations

6.1. The Deprecation Response Header Field

The "Deprecation" response header should be added to the permanent registry of message header fields (see [[RFC3864](#)]), taking into account the guidelines given by HTTP/1.1 [[RFC7231](#)].

Header Field Name: Deprecation

Applicable Protocol: Hypertext Transfer Protocol (HTTP)

Status: Standard

Author: Sanjay Dalal <sanjay.dalal@cal.berkeley.edu>,
Erik Wilde <erik.wilde@dret.net>

Change controller: IETF

Specification document: this specification,
[Section 2](#) "The Deprecation HTTP Response Header Field"

6.2. The Deprecation Link Relation Type

The "deprecation" link relation type should be added to the permanent registry of link relation types according to [Section 4.2 of \[RFC8288\]](#):

Relation Type: deprecation

Applicable Protocol: Hypertext Transfer Protocol (HTTP)

Status: Standard

Author: Sanjay Dalal <sanjay.dalal@cal.berkeley.edu>,
Erik Wilde <erik.wilde@dret.net>

Change controller: IETF

Specification document: this specification,
[Section 3](#) "The Deprecation Link Relation Type"

7. Implementation Status

Note to RFC Editor: Please remove this section before publication.

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in [\[RFC7942\]](#). The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

According to [RFC 7942](#), "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation and feedback that have made the implemented protocols more mature. It is up to the individual working groups to use this information as they see fit".

Organization: Zapier

Description: Zapier uses two custom HTTP headers named "X-API-Deprecation-Date" and "X-API-Deprecation-Info"

Reference: <https://zapier.com/engineering/api-geriatrics/>

Organization: IBM

IBM uses a custom HTTP header named "Deprecated"

Reference:

https://www.ibm.com/support/knowledgecenter/en/SS42VS_7.3.1/com.ibm.qradar.doc/c_rest_api_getting_started.html

Organization: Ultipro

Description: Ultipro uses the HTTP "Warning" header as described in [Section 5.5 of \[RFC7234\]](#) with code "299"

Reference: <https://connect.ultipro.com/api-deprecation>

Organization: Clearbit

Description: Clearbit uses a custom HTTP header named "X-API-Warn"

Reference: <https://blog.clearbit.com/dealing-with-deprecation/>

Organization: PayPal

Description: PayPal uses a custom HTTP header named "PayPal-Deprecated"

Reference: <https://github.com/paypal/api-standards/blob/master/api-style-guide.md#runtime>

8. Security Considerations

The content of a "Link" header field is not secure, private or integrity-guaranteed, and due caution should be exercised when using it. Use of Transport Layer Security (TLS) with HTTP ([RFC7230]) is currently the only end-to-end way to provide such protection.

The suggested "Link" header fields make extensive use of IRIs and URIs. See [RFC3987] for security considerations relating to IRIs. See [RFC3986] for security considerations relating to URIs. See [RFC7230] for security considerations relating to HTTP headers.

Applications that take advantage of typed links should consider the attack vectors opened by automatically following, trusting, or otherwise using links gathered from the HTTP headers. In particular, Link headers that use the "successor-version", "latest-version" or "alternate" relation types should be treated with due caution. See [RFC5829] for security considerations relating to these link relation types.

9. Example

Just deprecation header without any Link headers.

```
Deprecation: version="v1"
```

Deprecation header with link to the successor version.

```
Deprecation: version="v1"
```

```
Link: <https://api.example.com/v2/customers>; rel="successor-version"
```

Deprecation header with links for the successor version and for the API developer's deprecation policy. Also, it shows sunset date for the deprecated version (v1).

```
Deprecation: version="v1"
```

```
Sunset: Fri, 11 Nov 2020 23:59:59 GMT
```

```
Link: <https://api.example.com/v2/customers>; rel="successor-version"
```

```
Link: <https://developer.example.com/deprecation>; rel="deprecation"
```

10. References

10.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC3864] Klyne, G., Nottingham, M., and J. Mogul, "Registration Procedures for Message Header Fields", [BCP 90](#), [RFC 3864](#), DOI 10.17487/RFC3864, September 2004, <<https://www.rfc-editor.org/info/rfc3864>>.
- [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, <<https://www.rfc-editor.org/info/rfc3986>>.
- [RFC3987] Duerst, M. and M. Suignard, "Internationalized Resource Identifiers (IRIs)", [RFC 3987](#), DOI 10.17487/RFC3987, January 2005, <<https://www.rfc-editor.org/info/rfc3987>>.
- [RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, [RFC 5234](#), DOI 10.17487/RFC5234, January 2008, <<https://www.rfc-editor.org/info/rfc5234>>.

- [RFC7230] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing", [RFC 7230](#), DOI 10.17487/RFC7230, June 2014, <<https://www.rfc-editor.org/info/rfc7230>>.
- [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, <<https://www.rfc-editor.org/info/rfc7231>>.
- [RFC7234] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Caching", [RFC 7234](#), DOI 10.17487/RFC7234, June 2014, <<https://www.rfc-editor.org/info/rfc7234>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in [RFC 2119](#) Key Words", [BCP 14](#), [RFC 8174](#), DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8288] Nottingham, M., "Web Linking", [RFC 8288](#), DOI 10.17487/RFC8288, October 2017, <<https://www.rfc-editor.org/info/rfc8288>>.

[10.2. Informative References](#)

- [Deprecation] Marks, S., "JEP 277 - Enhanced Deprecation", 2017, <<http://openjdk.java.net/jeps/277>>.
- [RFC5829] Brown, A., Clemm, G., and J. Reschke, Ed., "Link Relation Types for Simple Version Navigation between Web Resources", [RFC 5829](#), DOI 10.17487/RFC5829, April 2010, <<https://www.rfc-editor.org/info/rfc5829>>.
- [RFC7942] Sheffer, Y. and A. Farrel, "Improving Awareness of Running Code: The Implementation Status Section", [BCP 205](#), [RFC 7942](#), DOI 10.17487/RFC7942, July 2016, <<https://www.rfc-editor.org/info/rfc7942>>.
- [Sunset] Wilde, E., "The Sunset HTTP Header Field", 2019, <<https://tools.ietf.org/html/draft-wilde-sunset-header-11>>.

[Appendix A. Acknowledgments](#)

The authors would like to thank Mark Nottingham and Nikhil Kolekar for reviewing this specification.

The authors take all responsibility for errors and omissions.

Authors' Addresses

Sanjay Dalal

Email: sanjay.dalal@cal.berkeley.edu

URI: <https://github.com/sdatspun2>

Erik Wilde

Email: erik.wilde@dret.net

URI: <http://dret.net/netdret>