

Network Working Group
Internet-Draft
Updates: 5545 (if approved)
Intended status: Standards Track
Expires: 18 September 2025

M. Douglass
17 March 2025

Calendar subscription upgrades
draft-ietf-calext-subscription-upgrade-13

Abstract

This specification updates RFC5545 to add the value DELETED to the STATUS property.

This specification also adds values to the Preferences Registry defined in RFC7240 to add the subscribe-enhanced-get and limit preferences and to the link relations directory defined in RFC8288.

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 18 September 2025.

Copyright Notice

Copyright (c) 2025 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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

| | |
|---|----|
| 1. Introduction | 3 |
| 1.1. Terms and Definitions | 3 |
| 2. Discovering alternative access methods | 3 |
| 3. Enhanced GET | 4 |
| 3.1. General | 4 |
| 3.2. Deletions | 5 |
| 3.3. Paging the response | 5 |
| 3.4. Caching of responses | 6 |
| 3.5. Examples | 6 |
| 4. Changes to the iCalendar specifications | 8 |
| 4.1. Redefined Status property | 8 |
| 5. Header Field: Sync-Token | 10 |
| 6. New Prefer header field preferences | 10 |
| 6.1. Preference subscribe-enhanced-get | 10 |
| 6.2. Preference limit | 10 |
| 7. Link relations | 10 |
| 7.1. General | 11 |
| 7.2. subscribe-caldav | 11 |
| 7.3. subscribe-caldav-auth | 11 |
| 7.4. subscribe-webdav-sync | 11 |
| 7.5. subscribe-enhanced-get | 11 |
| 8. Security Considerations | 12 |
| 9. Privacy Considerations | 12 |
| 10. IANA Considerations | 12 |
| 10.1. Sync-Token HTTP Header Field Registration | 12 |
| 10.2. Preference Registrations | 12 |
| 10.3. Link Relation Registrations | 13 |
| 10.4. New and updated iCalendar Elements Registration | 13 |
| 10.4.1. Initialization of the Status registry | 13 |
| 10.4.2. Update of the Status registry | 14 |
| 11. Acknowledgements | 14 |
| 12. Normative References | 14 |
| Author's Address | 16 |

1. Introduction

Currently, clients subscribe to calendar feeds as an iCalendar file which is often published as a resource accessible using the unofficial 'webcal' URI scheme.

The only available option for updating that resource is the usual HTTP polling of cached resources using Section 8.8.2 of [RFC9110] Last-Modified or Section 8.8.3 of [RFC9110] Etag.

There is the usual tension between clients wishing to see a timely response to changes and servers not wishing to be overloaded by frequent requests for possibly large amounts of data.

This specification introduces an approach whereby clients can discover a more performant access method. Given the location of the resource as an iCalendar file, the client can perform a HEAD request on the resource and inspect the returned headers which will offer a number of alternative access methods.

Given that many clients and servers already support CalDAV this provides an easy upgrade path for those clients. Additionally, an enhanced GET protocol is specified here to allow a lightweight implementation.

The use of subscription upgrade may help reduce load on servers, but perhaps more importantly it allows mobile devices to use a more efficient update mechanism, reducing data transferred and presumably improving battery life.

1.1. Terms and Definitions

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.

Additionally, the rule for URI is included from [RFC3986].

2. Discovering alternative access methods

The advertising of other access points is achieved through the use of the LINK header as defined in [RFC8288]. New link relation types are defined in this specification - each being associated with a protocol or protocol subset.

These LINK headers will be delivered when a client carries out a HEAD request targeting the URL of the resource.

EXAMPLE

This is an example of a HEAD request and the response from a server that supports the enhanced GET method.

>> Request <<

```
HEAD /caldata/events.ics HTTP/1.1
Host: example.com
Accept: text/calendar
```

>> Response <<

```
HTTP/1.1 200 OK
Content-Length: xxxx
Link: <https://example.com/subscribe/events.ics>;
      rel="subscribe-enhanced-get"
```

Note that the target for an upgraded service may be the same as for the initial resource.

3. Enhanced GET

3.1. General

This is a lightweight protocol which allows simple clients to efficiently discover and download changes in the targeted resource.

It has many similarities to [RFC6578] WebDAV sync and for a server could be implemented as an extension of the specification.

In this protocol the client MUST include the Prefer header field preference "subscribe-enhanced-get". If a sync token is available it is passed as a Sync-Token header field.

The resource is treated as a set of individual events each of which may be updated or deleted separately. Note that a recurring event and its overrides are treated as a single entity. The client will first fetch the entire iCalendar file. On subsequent requests it uses the Prefer header field and a Sync-Token header field to indicate that it wants a set of changes since the last fetch.

If no Sync-Token header field is supplied the server MUST respond with a full set of data and SHOULD set the Preference-Applied header field and a new Sync-Token header field value.

If the token is valid, it SHOULD return with a set of changed entities and MUST set the Preference-Applied header field and a new Sync-Token header field value.

When a server receives an invalid token it MUST return a 409 status (Conflict). The server MAY choose to return an error message in the body.

The client MUST respond to this error by discarding the cached data and restarting the interaction from scratch, i.e. retrieve the full set of data then poll for updates.

In the absence of a Prefer header field, the server MAY include the Sync-Token header field. This can act as a signal to the client that the server supports the enhanced-get protocol.

3.2. Deletions

When an entity (VEVENT, VTODO or other valid top-level component) is deleted from the source data the server needs to be able to inform a client of the deletion. This specification introduces a new value "DELETED" for the Section 3.8.1.11 of [RFC5545] STATUS property.

On the first enhanced GET after the entity has been deleted a skeleton, but valid, entity will be returned with STATUS: DELETED. The receiving client is free to remove the entity or update its STATUS property.

On subsequent fetches the entity will not be returned.

As noted above, the skeleton entity returned must be a valid component as defined by the specifications. For example, a VEVENT MUST contain the DTSTAMP, DTSTART and UID properties as well as the STATUS: DELETED property. The DTSTAMP and DTSTART may be generated and the UID MUST be the UID of the deleted entity.

3.3. Paging the response

A client may explicitly request a limit on the size of the response by specifying the Prefer header field preference "limit=n" where n is the number of components.

When a server receives a request specifying such a limit it SHOULD limit the response to that number of components. If the limit causes a truncation in the response the server should respond with a Preference-Applied header specifying the limit that was applied and return a sync token which may be used to retrieve the next batch of data.

This allows the client to immediately resubmit a request for the next batch using the updated token.

A server MAY choose to limit the response size. The behavior MUST be as if the client had provided a preference for that size - allowing the client to retrieve the full set of data in batches.

Note that the process above assumes that the current position in the resource is encoded in some way within the sync token.

3.4. Caching of responses

To enable proper caching of responses the server SHOULD provide a Vary header field defined in Section 7.1.4 of [RFC7231] in responses that names the Prefer and Sync-Token header fields along with any other that are appropriate.

See Section 2 of [RFC7240] for a brief comment on use of Vary and Prefer.

3.5. Examples

EXAMPLE 1

This is an example of the initial request and response from a server that supports the enhanced GET method. Note the use of the Vary header so a caching proxy can key off the client's Sync-Token and preference.

>> Request <<

```
GET /events.ics HTTP/1.1
Host: example.com
Accept: text/calendar
Prefer: subscribe-enhanced-get
```

>> Response <<

```
HTTP/1.1 200 OK
Content-Length: xxxx
Sync-Token: "data:,1234567"
Preference-Applied: subscribe-enhanced-get
Vary: Prefer, Sync-Token
```

```
BEGIN:VCALENDAR:
? /* full feed */
END:VCALENDAR
```

EXAMPLE 2

This is an example of the subsequent request and response when no changes have occurred.

>> Request <<

```
GET /events.ics HTTP/1.1
Host: example.com
Accept: text/calendar
Prefer: subscribe-enhanced-get
Sync-Token: "data:,1234567"
```

>> Response <<

```
HTTP/1.1 304 Not Modified
Content-Length: 0
Sync-Token: "data:,1234567"
Preference-Applied: subscribe-enhanced-get
Vary: Prefer, Sync-Token
```

EXAMPLE 3

This is an example of the subsequent request and response for an old or invalid token.

>> Request <<

```
GET /events.ics HTTP/1.1
Host: example.com
Accept: text/calendar
Sync-Token: "data:,1234567"
Prefer: subscribe-enhanced-get
```

>> Response <<

```
HTTP/1.1 409 Conflict
Content-Length: xxxx
Preference-Applied: subscribe-enhanced-get
```

EXAMPLE 4

This is an example of the subsequent request and response when changes have occurred.

>> Request <<

```
GET /events.ics HTTP/1.1
Host: example.com
Accept: text/calendar
Sync-Token: "data:,1234567"
Prefer: subscribe-enhanced-get
```

>> Response <<

```
HTTP/1.1 200 OK
Content-Type: text/calendar
Vary: Prefer, Sync-Token
Sync-Token: "data:,4567890"
Preference-Applied: subscribe-enhanced-get
```

```
BEGIN:VCALENDAR:
... only new/changed events
... deleted events have STATUS:DELETED
END:VCALENDAR
```

4. Changes to the iCalendar specifications

This specification updates [RFC5545] to add the value DELETED to the STATUS property.

4.1. Redefined Status property

Property name STATUS

Purpose This property defines the overall status or confirmation for the calendar component.

Value Type TEXT

Property Parameters IANA and non-standard property parameters can be specified on this property.

Conformance This property can be specified once in "VEVENT", "VTOD0", or "VJOURNAL" calendar components.

Description In a group-scheduled calendar component, the property is used by the "Organizer" to provide a confirmation of the event to the "Attendees". For example in a "VEVENT" calendar component, the "Organizer" can indicate that a meeting is tentative, confirmed, or cancelled. In a "VTOD0" calendar component, the "Organizer" can indicate that an action item needs action, is completed, is in process or being worked on, or has been

cancelled. In a "VJOURNAL" calendar component, the "Organizer" can indicate that a journal entry is draft, final, or has been cancelled or removed.

Format Definition

This property is defined by the following notation:

```
status          = "STATUS" statparam ":" statvalue CRLF
statparam       = *(";" other-param)
statvalue       = (statvalue-event
                  /  statvalue-todo
                  /  statvalue-jour)

statvalue-event = "TENTATIVE"      ;Indicates event is tentative.
                  / "CONFIRMED"    ;Indicates event is definite.
                  / "CANCELLED"    ;Indicates event was cancelled.
                  / "DELETED"      ;Indicates event was deleted.
;Status values for a "VEVENT"

statvalue-todo  = "NEEDS-ACTION" ;Indicates to-do needs action.
                  / "COMPLETED"  ;Indicates to-do completed.
                  / "IN-PROCESS"  ;Indicates to-do in process of.
                  / "CANCELLED"   ;Indicates to-do was cancelled.
                  / "DELETED"     ;Indicates to-do was deleted.
;Status values for "VTODO".

statvalue-jour  = "DRAFT"         ;Indicates journal is draft.
                  / "FINAL"       ;Indicates journal is final.
                  / "CANCELLED"   ;Indicates journal is removed.
                  / "DELETED"     ;Indicates journal was deleted.
;Status values for "VJOURNAL".
```

Example

EXAMPLE 1

The following is an example of this property for a "VEVENT" calendar component:

```
STATUS:TENTATIVE
```

EXAMPLE 2

The following is an example of this property for a "VTODO" calendar component:

STATUS:NEEDS-ACTION

EXAMPLE 3

The following is an example of this property for a "VJOURNAL" calendar component:

STATUS:DRAFT

5. Header Field: Sync-Token

This specification defines a new header field Sync-Token for use by the enhanced GET method.

Sync-Token = DQUOTE URI DQUOTE

The value MUST be a URI which could be a data URI. The synchronization token itself MUST be treated as an "opaque" string by the client, i.e., the actual string data has no specific meaning or syntax. These restrictions align the specification with Section 3.2 of [RFC6578].

EXAMPLE

This is an example of the Sync-Token header field:

Sync-Token: "data:,1234567"

6. New Prefer header field preferences

6.1. Preference subscribe-enhanced-get

This indicates that the client expects the server to handle the GET method according to the specifications for enhanced get.

pref-subscribe-enhanced-get = "subscribe-enhanced-get"

6.2. Preference limit

This preference parameter provides a limit on the number of components returned for enhanced get.

pref-limit = "limit" BWS "=" BWS 1*DIGIT

7. Link relations

7.1. General

This clause defines a number of new link relations required to facilitate subscription upgrades.

7.2. subscribe-caldav

This specifies an access point which is a full implementation of caldav but requires no authentication. The end point allows the full range of reports as defined by the CalDAV specification.

The client MUST follow the specification to determine exactly what operations are allowed on the access point - for example to determine if DAV:sync-collection is supported.

The URL MAY include some form of token to allow write access to the targeted collection. The client must check its permissions to determine whether it has been granted write access.

7.3. subscribe-caldav-auth

This specifies an access point which is a full implementation of caldav and requires authentication. This may allow read-write access to the resource.

The client MUST follow the specification to determine exactly what operations are allowed on the access point - for example to determine if DAV:sync-collection is supported.

7.4. subscribe-webdav-sync

This specifies an access point which supports only [RFC6578] (WebDAV sync).

This allows the client to issue a DAV:sync-collection report on the resource to obtain updates.

The client MUST follow the requirements of [RFC6578].

7.5. subscribe-enhanced-get

This specifies an access point which supports the enhanced GET protocol defined in Section 3.

The client and server MUST follow the protocol as defined in Section 3.

8. Security Considerations

Applications using these properties need to be aware of the risks entailed in using the URIs provided as values. See [RFC3986] for a discussion of the security considerations relating to URIs.

9. Privacy Considerations

Properties with a "URI" value type can expose their users to privacy leaks as any network access of the URI data can be tracked. Clients SHOULD NOT automatically download data referenced by the URI without explicit instruction from users. This specification does not introduce any additional privacy concerns beyond those described in [RFC5545].

10. IANA Considerations

10.1. Sync-Token HTTP Header Field Registration

This specification updates the "Message Headers" registry entry for "Sync-Token" in [RFC3864] to refer to this document.

Header Field Name: Sync-Token
Protocol: http
Status: standard
Reference: <this-document>

10.2. Preference Registrations

The following preferences have been added to the HTTP Preferences Registry defined in [RFC7240]

Preference subscribe-enhanced-get

Value None.

Description Marks the interaction as enhanced get and provides the optional sync-token and page size.

Reference this document

Preference limit

Value An integer page size.

Description Provide a limit on the number of components in the response.

Reference this document

10.3. Link Relation Registrations

The following link relation values have been added to the Reference Types Registry defined in Section 6.2.2 of [RFC8288]:

| Relation Name | Description | Reference |
|------------------------|-------------|-------------|
| subscribe-caldav | Current | Section 7.2 |
| subscribe-caldav_auth | Current | Section 7.3 |
| subscribe-webdav-sync | Current | Section 7.4 |
| subscribe-enhanced_get | Current | Section 7.5 |

Table 1

10.4. New and updated iCalendar Elements Registration

This specification updates [RFC5545] by adding and updating a number of elements according to the procedures and templates specified in Section 8.2 of [RFC5545].

10.4.1. Initialization of the Status registry

This specification updates [RFC5545] by adding a Status value registry to the iCalendar Elements registry located here: <https://www.iana.org/assignments/icalendar> and initializing it as per [RFC5545].

| Name | Status | Reference |
|--------------|---------|-------------------------------|
| CANCELLED | Current | Section 3.8.1.11 of [RFC5545] |
| COMPLETED | Current | Section 3.8.1.11 of [RFC5545] |
| CONFIRMED | Current | Section 3.8.1.11 of [RFC5545] |
| DRAFT | Current | Section 3.8.1.11 of [RFC5545] |
| FINAL | Current | Section 3.8.1.11 of [RFC5545] |
| IN-PROCESS | Current | Section 3.8.1.11 of [RFC5545] |
| NEEDS-ACTION | Current | Section 3.8.1.11 of [RFC5545] |
| TENTATIVE | Current | Section 3.8.1.11 of [RFC5545] |

Table 2: Initial Status Value Registry

10.4.2. Update of the Status registry

This specification further updates the Status registry with the additional DELETED value defined in this document.

| Value | Status | Reference |
|---------|---------|------------------------|
| DELETED | Current | This Spec, Section 3.2 |

Table 3: Updated Status Value Registry

11. Acknowledgements

The author would like to thank the members of the CalConnect Calendar Sharing technical committee and the following individuals for contributing their ideas and support:

Marten Gajda, Ken Murchison, Garry Shutler

12. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", IETF, DOI 10.17487/RFC2119, BCP 14, RFC 2119, March 1997,
<<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC3864] Klyne, G., Nottingham, M., and J. Mogul, "Registration Procedures for Message Header Fields", IETF, DOI 10.17487/RFC3864, BCP 90, RFC 3864, September 2004,
<<https://www.rfc-editor.org/info/rfc3864>>.
- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", IETF, STD 66, DOI 10.17487/RFC3986, BCP 66, RFC 3986, January 2005,
<<https://www.rfc-editor.org/info/rfc3986>>.
- [RFC5545] Desruisseaux, B., "Internet Calendaring and Scheduling Core Object Specification (iCalendar)", IETF, DOI 10.17487/RFC5545, RFC 5545, September 2009,
<<https://www.rfc-editor.org/info/rfc5545>>.
- [RFC6578] Daboo, C. and A. Quillaud, "Collection Synchronization for Web Distributed Authoring and Versioning (WebDAV)", IETF, DOI 10.17487/RFC6578, RFC 6578, March 2012,
<<https://www.rfc-editor.org/info/rfc6578>>.
- [RFC7231] Fielding, R. and J. Reschke, "Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content", IETF, DOI 10.17487/RFC7231, RFC 7231, June 2014,
<<https://www.rfc-editor.org/info/rfc7231>>.
- [RFC7240] Snell, J., "Prefer Header for HTTP", IETF, DOI 10.17487/RFC7240, RFC 7240, June 2014,
<<https://www.rfc-editor.org/info/rfc7240>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", IETF, DOI 10.17487/RFC8174, BCP 14, RFC 8174, May 2017,
<<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8288] Nottingham, M., "Web Linking", IETF, DOI 10.17487/RFC8288, RFC 8288, October 2017,
<<https://www.rfc-editor.org/info/rfc8288>>.
- [RFC9110] Fielding, R., Nottingham, M., and J. Reschke, "HTTP Semantics", IETF, STD 97, DOI 10.17487/RFC9110, BCP 97, RFC 9110, June 2022,
<<https://www.rfc-editor.org/info/rfc9110>>.

Internet-Draft

Calendar subscription upgrades

March 2025

Author's Address

Michael Douglass

Email: mdouglass@bedework.com

Douglass

Expires 18 September 2025

[Page 16]