☰ Table of Contents ▲

Workgroup:	Network Working Group
Internet-Draft:	draft-ietf-calext-subscription-upgrade-02
:	
Updates:	5988 (if approved)
Published:	29 July 2020
Intended Status:	Standards Track
Expires:	30 January 2021
Author:	M. Douglass

Calendar subscription upgrades

Abstract

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

This specification also updates [RFC7240] to add the subscribe- enhanced-get and limit preferences.

Status of This Memo

Copyright Notice

1. Acknowledgements

The author would also 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

The authors would also like to thank CalConnect, the Calendaring and Scheduling Consortium, for advice with this specification.

2. Introduction

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

The only available option for updating that resource is the usual HTTP polling of cached resources using Etags.

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 perfom 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. CalDAV and DAV subsets are specified here to allow lighter weight implementations.

2.1. 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].

3. Discovering alternative access methods

The advertising of other access points is achieved through the use of the LINK header as defined in [RFC5988]. 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.

```
    &gt;&gt; Request &lt;&lt;

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

    &gt;&gt; Response &lt;&lt;

    HTTP/1.1 200 OK
    Content-Length: xxxx
    Link: &lt;http://example.com/subscribe/events.ics&gt;;
                 rel="subscribe-enhanced-get"
```

Figure 1

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

4. Enhanced GET

4.1. 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 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. 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 SHOULD respond with a full set of data. Otherwise, if the token is valid, it SHOULD return with a set of changed entities.

In both cases the server should set the Preference-Applied header field and a new Sync-Token header field value.

4.2. 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 for the STATUS property of DELETED.

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 it's STATUS property.

On subsequent fetches the entity will not be returned.

4.3. Handling of invalid sync tokens

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 SHOULD respond to this error by restarting the interaction from scratch, i.e. retrieve the full set of data then poll for updates.

4.4. 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 SHOULD be as if the client had provided a preference for that size - allowing the client to retrieve the full set of data in batches.

4.5. Caching of responses

To enable proper caching of responses the server SHOULD provide a VARY header field in responses that names the Prefer and Sync-Token header fields along with any other that are appropriate.

Clients should order the preferences as following so that identical responses can be identified:

- subscribe-enhanced-get
- limit

4.6. 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.

```
    &gt;&gt; Request &lt;&lt;

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

    &gt;&gt; Response &lt;&lt;

    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
```

Figure 2

EXAMPLE 2

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

```
    &gt;&gt; Request &lt;&lt;

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

    &gt;&gt; Response &lt;&lt;

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

Figure 3

EXAMPLE 3

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

```
&gt;&gt; Request &lt;&lt;

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

    &gt;&gt; Response &lt;&lt;

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

Figure 4

EXAMPLE 4

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

```
&gt;&gt; Request &lt;&lt;

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

    &gt;&gt; Response &lt;&lt;

    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
```

Figure 5

5. Changes to the iCalendar specifications

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

5.1. 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", "VTODO", 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 "VTODO" 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".
```

Figure 6

Example

EXAMPLE 1

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

```
STATUS:TENTATIVE
```

Figure 7

EXAMPLE 2

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

```
STATUS:NEEDS-ACTION
```

Figure 8

EXAMPLE 3

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

```
STATUS:DRAFT
```

Figure 9

6. Header Field: Sync-Token

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

```
    Accept = DQUOTE URI DQUOTE
```

Figure 10

The value MUST be a URI. This will generally be a data URI representing an opaque token. Client MUST not attempt to interpret the data URI value.

EXAMPLE

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

```
    Sync-Token: "data:,1234567"
```

Figure 11

7. New Prefer header field preferences

7.1. 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"
```

Figure 12

7.2. Preference limit

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

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

Figure 13

8. Link relations

8.1. General

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

8.2. 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 sync-report is supported.

The URL MAY include some form of token to allow write access to the targeted collection. The client must check it's permissions to determine whether or not it has been granted write access.

8.3. 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 sync-report is supported.

8.4. subscribe-webdav-sync

This specifies an access point which supports only webdav sync.

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

The client MUST follow that specification.

8.5. subscribe-enhanced-get

This specifies an access point which supports something new.

The client MUST follow that specification.

9. 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. == 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>
```

Figure 14

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

This document defines the following new iCalendar properties to be added to the registry defined in [RFC5545]:

Relation Name	Description	Reference
subscribe-caldav	Current	Section 8.2
subscribe-caldav_auth	Current	Section 8.3
subscribe-webdav-sync	Current	Section 8.4
subscribe-enhanced_get	Current	Section 8.5

Table 1

11. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", IETF RFC 2119, IETF 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", IETF RFC 3864, IETF 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", IETF RFC 3986, IETF RFC 3986, DOI 10.17487/RFC3986, January 2005, <https://www.rfc-editor.org/info/rfc3986>.

[RFC5545]

Desruisseaux, B., Ed., "Internet Calendaring and Scheduling Core Object

[RFC5988] Nottingham, M., "Web Linking", IETF RFC 5988, IETF RFC 5988, DOI 10.17487/RFC5988, October 2010, <https://www.rfc-editor.org/info/rfc5988>.

[RFC7240] Snell, J., "Prefer Header for HTTP", IETF RFC 7240, IETF RFC 7240, DOI 10.17487/RFC7240, June 2014, <https://www.rfc-editor.org/info/rfc7240>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", IETF RFC 8174, IETF RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

Appendix A. Open issues

Vary Ensure we get that right.

Appendix B. Change log

calext00 2019-06-05 MD

- First calext version
- Use Sync-Token header rather than parameter

v04 2019-03-07 MD

- Reference to RFC 6538 - WebDAV sync and RFC 7240 Prefer
- Go back to HEAD
- New Preference and parameters.
- Examples
- More text for extended get. Talk about deletions.

v01 2017-02-17 MD

- Add text about OPTIONS
- Add text abut read/write CalDAV

v00 2017-02-15 MD

- First pass

Author's Address

Michael Douglass
Email: mdouglass@bedework.com