

# Calendaring and scheduling – Calendar subscription upgrades

Committee Draft Standard

## **Warning for drafts**

This document is not a CalConnect Standard. It is distributed for review and comment, and is subject to change without notice and may not be referred to as a Standard. Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

© 2018 The Calendaring and Scheduling Consortium, Inc.

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from the address below.

The Calendaring and Scheduling Consortium, Inc.

4390 Chaffin Lane  
McKinleyville  
California 95519  
United States of America

[copyright@calconnect.org](mailto:copyright@calconnect.org)  
[www.calconnect.org](http://www.calconnect.org)

# Contents

<b>Foreword</b> .....	<b>iv</b>
<b>Introduction</b> .....	<b>v</b>
<b>Acknowledgements</b> .....	<b>vi</b>
<b>1. Scope</b> .....	<b>1</b>
<b>2. Normative References</b> .....	<b>1</b>
<b>3. Discovering alternative access methods</b> .....	<b>1</b>
<b>4. Enhanced GET</b> .....	<b>2</b>
<b>4.1. General</b> .....	<b>2</b>
<b>4.2. Deletions</b> .....	<b>2</b>
<b>4.3. Handling of invalid sync tokens</b> .....	<b>2</b>
<b>4.4. Paging the response</b> .....	<b>2</b>
<b>4.5. Caching of responses</b> .....	<b>3</b>
<b>4.6. Examples</b> .....	<b>3</b>
<b>5. Changes to the iCalendar specifications</b> .....	<b>4</b>
<b>5.1. Redefined Status property</b> .....	<b>4</b>
<b>6. Header Field: Sync-Token</b> .....	<b>6</b>
<b>7. New Prefer header field preferences</b> .....	<b>6</b>
<b>7.1. Preference subscribe-enhanced-get</b> .....	<b>6</b>
<b>7.2. Preference limit</b> .....	<b>6</b>
<b>8. Link relations</b> .....	<b>7</b>
<b>8.1. General</b> .....	<b>7</b>
<b>8.2. subscribe-caldav</b> .....	<b>7</b>
<b>8.3. subscribe-caldav-auth</b> .....	<b>7</b>
<b>8.4. subscribe-webdav-sync</b> .....	<b>7</b>
<b>8.5. subscribe-enhanced-get</b> .....	<b>7</b>
<b>9. Security Considerations</b> .....	<b>7</b>
<b>10. IANA Considerations</b> .....	<b>8</b>
<b>10.1.Sync-Token HTTP Header Field Registration</b> .....	<b>8</b>
<b>10.2.Preference Registrations</b> .....	<b>8</b>
<b>10.3.Link Relation Registrations</b> .....	<b>8</b>
<b>Annex A. (informative) Open issues</b> .....	<b>9</b>
<b>Annex B. (informative) Change log</b> .....	<b>10</b>
<b>Bibliography</b> .....	<b>11</b>

## Foreword

This specification updates [RFC 5545](#) to add the value DELETED to the STATUS property.

This specification also updates [RFC 7240](#) to add the subscribe- enhanced-get and limit preferences.

The Calendaring and Scheduling Consortium (“CalConnect”) is a global non-profit organization with the aim to facilitate interoperability of collaborative technologies and tools through open standards.

CalConnect works closely with international and regional partners, of which the full list is available on our website (<https://www.calconnect.org/about/liaisons-and-relationships>).

The procedures used to develop this document and those intended for its further maintenance are described in the CalConnect Directives.

In particular the different approval criteria needed for the different types of CalConnect documents should be noted. This document was drafted in accordance with the editorial rules of the CalConnect Directives.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CalConnect shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be provided in the Introduction.

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

This document was prepared by Technical Committee *CALENDAR*.

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

## **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.

# Calendaring and scheduling – Calendar subscription upgrades

## 1. Scope

This document provides a mechanism to allow subscribers to calendar feeds to upgrade to a more performant protocol.

## 2. Normative References

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

RFC 2518, *HTTP Extensions for Distributed Authoring – WEBDAV*

RFC 3864, *Registration Procedures for Message Header Fields*

RFC 3986, *Uniform Resource Identifier (URI): Generic Syntax*

RFC 4791, *Calendaring Extensions to WebDAV (CalDAV)*

RFC 5545, *Internet Calendaring and Scheduling Core Object Specification (iCalendar)*

RFC 5546, *iCalendar Transport-Independent Interoperability Protocol (iTIP)*

RFC 6047, *iCalendar Message-Based Interoperability Protocol (iMIP)*

RFC 6638, *Scheduling Extensions to CalDAV*

RFC 5988, *Web Linking*

RFC 7240, *Prefer Header for HTTP*

## 3. Discovering alternative access methods

The advertising of other access points is achieved through the use of the LINK header as defined in [RFC 5988](#). 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: <http://example.com/subscribe/events.ics>;
      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 its 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.

```
>> 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
```

**Figure 2**

##### 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

**Figure 3**

**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
```

**Figure 4**

**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
```

**Figure 5**

## 5. Changes to the iCalendar specifications

This specification updates [RFC 5545](#) 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 its 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 [RFC 3986](#) 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 [RFC 5545](#).

## 10. IANA Considerations

### 10.1. Sync-Token HTTP Header Field Registration

This specification updates the “Message Headers” registry entry for “Sync-Token” in [RFC 3864](#) to refer to this document.

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

**Figure 1**

### 10.2. Preference Registrations

The following preferences have been added to the HTTP Preferences Registry defined in [RFC 7240](#)

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 [RFC 5545, section 8.2.3](#):

**Table 1**

Relation Name	Description	Reference
subscribe-caldav	Current	<a href="#">Clause 8.2</a>
subscribe-caldav_auth	Current	<a href="#">Clause 8.3</a>
subscribe-webdav-sync	Current	<a href="#">Clause 8.4</a>
subscribe-enhanced_get	Current	<a href="#">Clause 8.5</a>

## **Annex A.. Open issues**

Vary

Ensure we get that right.

## **Annex 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 about read/write CalDAV

v00 2017-02-15 MD

- First pass

## **Bibliography**