

Streaming Calendar and Contacts Data

Working 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
www.calconnect.org

Contents

Abstract.....	iv
Introduction.....	v
1. Scope.....	1
2. Normative references.....	1
3. Terms and definitions.....	1
4. Conventions.....	1
5. Connections.....	2
5.1. Introduction.....	2
5.2. Virtual Connections.....	2
5.3. Examples.....	2
6. Changes to the iCalendar specifications.....	4
6.1. Redefined Status property.....	4
7. Discovering alternative access methods.....	5
7.1. Link relation subscribe-caldav.....	6
7.2. Link relation subscribe-caldav-auth.....	6
7.3. Link relation subscribe-webdav-sync.....	6
7.4. Link relation subscribe-enhanced-get.....	6
8. Security Considerations.....	6
9. Privacy Considerations.....	6
10. IANA Considerations.....	7
10.1.Link Relation Registrations.....	7
11. Acknowledgements.....	7

Abstract

This specification defines a model for streaming calendar and contact data. This allows for a more efficient method for synchronizing collections of data and may be used to augment existing protocols such as CalDAV.

Introduction

Currently clients or servers maintain a synchronized copy of external data through subscriptions and polling or by the use of protocol extensions such as DAV Sync Report.

These methods in essence require polling at more or less regular intervals to detect changes. For more timely detection of changes, frequent polling is required. This leads to inefficiencies and costs in battery and network usage.

The situation may be improved by the use of push notifications but this adds complications to the system.

This specification introduces an approach whereby a clients opens streams and then simply writes to or reads from the stream. This is not conceptually a new invention. Messaging systems already provide such features. What this specification provides is the data structures which can be passed as messages and defines the allowable operations.

This specification will provide an approach to implementations using websockets or currently existing messaging systems such as JMS.

Streaming Calendar and Contacts Data

1. Scope

This specification defines a model for streaming calendar and contact data. This allows for a more efficient method for synchronizing collections of data and may be used to augment existing protocols such as CalDAV.

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.

IETF RFC 2119, BRADNER, S. *Key words for use in RFCs to Indicate Requirement Levels*. 1997. RFC Publisher. <https://www.rfc-editor.org/info/rfc2119>.

IETF RFC 2434, NARTEN, T., H. ALVESTRAND and Internet Engineering Task Force. *Guidelines for Writing an IANA Considerations Section in RFCs*. 1998. RFC Publisher. <https://www.rfc-editor.org/info/rfc2434>.

IETF RFC 3688, MEALLING, M. *The IETF XML Registry*. 2004. RFC Publisher. <https://www.rfc-editor.org/info/rfc3688>.

IETF RFC 3986, BERNERS-LEE, T., R. FIELDING and L. MASINTER. *Uniform Resource Identifier (URI): Generic Syntax*. 2005. RFC Publisher. <https://www.rfc-editor.org/info/rfc3986>.

IETF RFC 4589, SCHULZRINNE, H., H. TSCHOFENIG and Internet Engineering Task Force. *Location Types Registry*. 2006. RFC Publisher. <https://www.rfc-editor.org/info/rfc4589>.

IETF RFC 5545, Internet Engineering Task Force (committee). *Internet Calendaring and Scheduling Core Object Specification (iCalendar)*. 2009. RFC Publisher. <https://www.rfc-editor.org/info/rfc5545>.

IETF RFC 5546, Internet Engineering Task Force (committee). *iCalendar Transport-Independent Interoperability Protocol (iTIP)*. 2009. RFC Publisher. <https://www.rfc-editor.org/info/rfc5546>.

IETF RFC 5988, NOTTINGHAM, M. *Web Linking*. 2010. RFC Publisher. <https://www.rfc-editor.org/info/rfc5988>.

W3C xml, World Wide Web Consortium. *Extensible Markup Language (XML) 1.0 (Fifth Edition)*. <https://www.w3.org/TR/xml/>.

Internet-Draft draft-ietf-calext-extensions-00, DABOO, Cyrus. *New Properties for iCalendar*. 2015. <https://datatracker.ietf.org/doc/html/draft-ietf-calext-extensions-00>.

3. Terms and definitions

No terms and definitions are listed in this document.

4. 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 [IETF RFC 2119](https://www.rfc-editor.org/info/rfc2119).

5. Connections

5.1. Introduction

This specification assumes that there are two virtual streams open for each established virtual connection: an output stream and an input stream.

All operations are carried out by either end writing a message on the output stream and receiving an acknowledgment on the input stream.

All messages **MUST** receive an acknowledgement within the defined timeout period or they or their acknowledgement will be assumed to have been lost. The sender will need to carry out any appropriate recovery.

Each message sent **MUST** include a unique (within the connection) identifier which will be returned in the acknowledgment.

5.2. Virtual Connections

The actual systems on which this is running may have many restrictions. Connections may be forcibly shut down after some idle time and there will be the usual issues with mobile devices and connectivity.

The virtual connection **MUST** hide these issues from the client. A connection will last until one end is deemed to have shut down the connection.

On subsequent conditional fetches the entity will not be returned.

5.3. Examples

EXAMPLE 1

This is an example of the initial request and response from a server that supports the extended GET protocol.

>> Request <<

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

>> Response <<

```
HTTP/1.1 200 OK
Content-Length: xxxx
ETag: "1234"                current ETag (for conditional GET)
Vary: Prefer, If-None-Match  so caching proxy can key off of
                             client's ETag (sync token) and preference
```

```
BEGIN:VCALENDAR:
... /* full feed */
```

END:VCALENDAR

EXAMPLE 2

This is an example of the subsequent request and response when no changes have occurred. The Accept header field indicates that a VPATCH format is most desirable but simple text/calendar is acceptable.

>> Request <<

```
GET /events.ics HTTP/1.1
Host: example.com
Accept: text/calendar; q=0.5, component=VPATCH, text/calendar;
If-None-Match: "1234"           conditional request
Prefer: return=minimal
```

>> Response <<

```
HTTP/1.1 304 Not Modified
Content-Length: 0
ETag: "1234"
Vary: Prefer, If-None-Match
```

EXAMPLE 3

This is an example of the subsequent request and response when changes have occurred and the server can create the minimal format.

>> Request <<

```
GET /events.ics HTTP/1.1
Host: example.com
Accept: text/calendar; q=0.5, component=VPATCH, text/calendar;
If-None-Match: "1234"           conditional request
Prefer: return=minimal
```

>> Response <<

```
HTTP/1.1 200 OK
Content-Type: text/calendar
Content-Length: xxxx
ETag: "5678"           current ETag (for conditional GET)
Preference-Applied: return=minimal signals to client that stream is
changes only
Vary: Prefer, If-None-Match so caching proxy can key off of
client's ETag (sync token) and preference
```

```
BEGIN:VCALENDAR:
... only new/changed events
... when not returning VPATCH, deleted events have STATUS:DELETED
END:VCALENDAR
```

EXAMPLE 4

This is an example of the subsequent request and response when changes have occurred and the server cannot create the minimal format — perhaps because of an old or invalid token. Note there is no Preference-Applied header field.

>> Request <<

```
GET /events.ics HTTP/1.1
```

```

Host: example.com
Accept: text/calendar; q=0.5, component=VPATCH, text/calendar;
If-None-Match: "1234"           conditional request
Prefer: return=minimal

>> Response <<

HTTP/1.1 200 OK
Content-Type: text/calendar
Content-Length: xxxx
ETag: "5678"           current ETag (for conditional GET)
Vary: Prefer, If-None-Match   so caching proxy can key off of
client's ETag (sync token) and preference

BEGIN:VCALENDAR:
... full set of data
END:VCALENDAR

```

6. Changes to the iCalendar specifications

This specification updates [IETF RFC 5545](#) to add the value DELETED to the STATUS property. It also introduces the use of some properties to provide more information about the resource, for example the time range it covers.

6.1. 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, cancelled, or deleted. In a "VTODO" calendar component, the "Organizer" can indicate that an action item needs action, is completed, is in process or being worked on, has been cancelled, or has been deleted. In a "VJOURNAL" calendar component, the "Organizer" can indicate that a journal entry is draft, final, cancelled, or deleted.
Format Definition	This property is defined by the following notation: <pre> status = "STATUS" statparam ":" statvalue CRLF statparam = *(";" other-param) statvalue = (statvalue-event / statvalue-todo / statvalue-jour) </pre>

```

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

```
STATUS:TENTATIVE
```

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

```
STATUS:NEEDS-ACTION
```

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

```
STATUS:DRAFT
```

7. Discovering alternative access methods

The advertising of other access points is achieved through the use of the LINK header as defined in [IETF 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 an OPTIONS request targeting the URL of the resource.

7.1. Link relation 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.

7.2. Link relation 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.

7.3. Link relation 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.

NOTE Initial startup may require using ics to populate and obtain initial token.

The client **MUST** follow that specification.

7.4. Link relation subscribe-enhanced-get

This specifies an access point which supports enhanced GET functionality as defined in this specification.

The client **MUST** follow that specification.

8. Security Considerations

Applications using these properties need to be aware of the risks entailed in using the URIs provided as values. See [IETF RFC 3986](#) 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 [IETF RFC 5545](#).

10. IANA Considerations

10.1. Link Relation Registrations

This document defines the following new iCalendar properties to be added to the registry defined in [IETF RFC 5545, Section 8.2.3](#):

Table 1

Relation Name	Description	Reference
subscribe-caldav	Current	RFCXXXX, Clause 7.1
subscribe-caldav-auth	Current	RFCXXXX, Clause 7.2
subscribe-webdav-sync	Current	RFCXXXX, Clause 7.3
subscribe-enhanced-get	Current	RFCXXXX, Clause 7.4

11. 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

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