

JMAP
Internet-Draft
Intended status: Standards Track
Expires: October 31, 2019

N. Jenkins
FastMail
M. Douglass
Spherical Cow Group
April 29, 2019

JMAP for Calendars
draft-ietf-jmap-calendars-00

Abstract

This document specifies a data model for synchronising calendar data with a server using JMAP.

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 October 31, 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	2
1.2.	The Date datatypes	3
1.3.	Terminology	3
1.4.	Addition to the capabilities object	3
2.	Calendars	3
2.1.	Calendar/get	5
2.2.	Calendar/changes	5
2.3.	Calendar/set	5
3.	Calendar events	5
3.1.	CalendarEvent/get	6
3.2.	CalendarEvent/changes	6
3.3.	CalendarEvent/set	6
3.4.	CalendarEvent/copy	6
3.5.	CalendarEvent/query	7
3.5.1.	Filtering	7
3.5.2.	Sorting	8
3.6.	CalendarEvent/queryChanges	8
4.	Security considerations	8
5.	IANA considerations	9
5.1.	JMAP capability registration for "calendars"	9
6.	Normative References	9
	Authors' Addresses	9

1. Introduction

JMAP ([I-D.ietf-jmap-core] - JSON Meta Application Protocol) is a generic protocol for synchronising data, such as mail, calendars or contacts, between a client and a server. It is optimised for mobile and web environments, and aims to provide a consistent interface to different data types.

This specification defines a data model for synchronising calendar data between a client and a server using JMAP.

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.

Type signatures, examples and property descriptions in this document follow the conventions established in section 1.1 of [I-D.ietf-jmap-core].

Object properties may also have a set of attributes defined along with the type signature. These have the following meanings:

- o `*server-set*`: Only the server can set the value for this property. The client **MUST NOT** send this property when creating a new object of this type.
- o `*immutable*`: The value **MUST NOT** change after the object is created.
- o `*default*`: (This is followed by a JSON value). The value that will be used for this property if it is omitted in an argument, or when creating a new object of this type.

Data types defined in the core specification are used in this document.

1.2. The Date datatypes

Where "LocalDate" is given as a type, it means a string in the same format as "Date", but with the `_time-offset_` omitted from the end. The interpretation in absolute time depends upon the time zone for the event, which may not be a fixed offset (for example when daylight saving time occurs). For example, `"2014-10-30T14:12:00"`.

1.3. Terminology

The same terminology is used in this document as in the core JMAP specification.

1.4. Addition to the capabilities object

The capabilities object is returned as part of the standard JMAP Session object; see the JMAP spec. Servers supporting `_this_` specification **MUST** add a property called `"urn:ietf:params:jmap:calendars"` to the capabilities object.

The value of this property is an empty object in both the JMAP session `_capabilities_` property and an account's `_accountCapabilities_` property.

2. Calendars

A Calendar is a named collection of events. All events are associated with one, and only one, calendar.

A `*Calendar*` object has the following properties:

- o `*id*`: "Id" (immutable; server-set) The id of the calendar.
- o `*name*`: "String" The user-visible name of the calendar. This may be any UTF-8 string of at least 1 character in length and maximum 255 octets in size.
- o `*color*`: "String" Any valid CSS color value. The color to be used when displaying events associated with the calendar. The color SHOULD have sufficient contrast to be used as text on a white background.
- o `*sortOrder*`: "UnsignedInt" (default: 0) Defines the sort order of calendars when presented in the client's UI, so it is consistent between devices. The number MUST be an integer in the range $0 \leq \text{sortOrder} < 2^{31}$. A calendar with a lower order should be displayed before a calendar with a higher order in any list of calendars in the client's UI. Calendars with equal order should be sorted in alphabetical order by name. The sorting should take into locale-specific character order convention.
- o `*isVisible*`: "Boolean" (default: true) Should the calendar's events be displayed to the user at the moment?
- o `*mayReadFreeBusy*`: "Boolean" (server-set) The user may read the free-busy information for this calendar. In JMAP terms, this means the user may use this calendar as part of a filter in a `_CalendarEvent/query_` call, however unless `"mayRead == true"`, the events returned for this calendar will only contain free-busy information, and be stripped of any other data. This property MUST be `"true"` if `_mayRead_` is `"true"`.
- o `*mayReadItems*`: "Boolean" (server-set) The user may fetch the events in this calendar. In JMAP terms, this means the user may use this calendar as part of a filter in a `_CalendarEvent/query_` call
- o `*mayAddItems*`: "Boolean" (server-set) The user may add events to this calendar. In JMAP terms, this means the user may call `_CalendarEvent/set_` to create new events in this calendar or move existing events into this calendar from another calendar. This property MUST be `"false"` if the account to which this calendar belongs has the `_isReadOnly_` property set to `"true"`.
- o `*mayModifyItems*`: "Boolean" (server-set) The user may edit events in this calendar by calling `_CalendarEvent/set_` with the `_update_` argument referencing events in this collection. This property MUST be `"false"` if the account to which this calendar belongs has the `_isReadOnly_` property set to `"true"`.

- o `*mayRemoveItems*`: "Boolean" (server-set) The user may remove events from this calendar by calling `_CalendarEvent/set_` with the `_destroy_` argument referencing events in this collection, or by updating their `_calendarId_` property to a different calendar. This property MUST be "false" if the account to which this calendar belongs has the `_isReadOnly_` property set to "true".
- o `*mayRename*`: "Boolean" (server-set) The user may rename the calendar. This property MUST be "false" if the account to which this calendar belongs has the `_isReadOnly_` property set to "true".
- o `*mayDelete*`: "Boolean" (server-set) The user may delete the calendar itself. This property MUST be "false" if the account to which this calendar belongs has the `_isReadOnly_` property set to "true".

2.1. Calendar/get

Standard `"/get"` method as described in [I-D.ietf-jmap-core] section 5.1. The `_ids_` argument may be "null" to fetch all at once.

2.2. Calendar/changes

Standard `"/changes"` method as described in [I-D.ietf-jmap-core] section 5.2.

2.3. Calendar/set

Standard `"/set"` method as described in [I-D.ietf-jmap-core] section 5.3.

A calendar MAY be deleted that is currently associated with one or more events. In this case, the events belonging to this calendar MUST also be deleted. Conceptually, this MUST happen prior to the calendar itself being deleted, and MUST generate a `*push*` event that modifies the state of the `_CalendarEvent_` type for the account.

3. Calendar events

A `*CalendarEvent*` object contains information about an event, or recurring series of events, that takes place at a particular time. It is a `JSEvent` object, as defined in [I-D.ietf-calext-jscalendar], with the following additional properties:

- o `*id*`: "Id" The id of the event. This property is immutable.
- o `*calendarId*`: "Id" The id of the calendar this event belongs to.

- o `*participantId*`: "Id|null" The id of the participant in the `_participants_` object which corresponds to the account this event is in.

3.1. CalendarEvent/get

Standard `"/get"` method as described in [I-D.ietf-jmap-core] section 5.1.

3.2. CalendarEvent/changes

Standard `"/changes"` method as described in [I-D.ietf-jmap-core] section 5.2

3.3. CalendarEvent/set

Standard `"/set"` method as described in [I-D.ietf-jmap-core] section 5.3.

When an event is created, updated or destroyed, the server MUST also ensure the following:

- o Any alerts are scheduled/cancelled correctly.
- o If there is a `_participantId_`, and the corresponding participant has a `_role_` of "owner":
 - * If an event is created/updated: send a REQUEST iMIP email with the event as an ICS attachment to all participants that are not "you".
 - * When an event is destroyed, if it is in the future, then email all participants other than you the appropriate iMIP email to inform them that the event has been cancelled. If it is in the past, the server SHOULD NOT send a message.
- o If there is a `_participantId_`, and the corresponding participant does not have a `_role_` of "owner", and the `_scheduleStatus_` is updated for this participant, send the appropriate iMIP email to the `_replyTo_` address.

3.4. CalendarEvent/copy

Standard `"/copy"` method as described in [I-D.ietf-jmap-core] section 5.4.

3.5. CalendarEvent/query

Standard `/query` method as described in [I-D.ietf-jmap-core] section 5.5.

3.5.1. Filtering

A `*FilterCondition*` object has the following properties:

- o `*inCalendars*`: `"Id[]|null"` A list of calendar ids. An event must be in ANY of these calendars to match the condition.
- o `*after*`: `"UTCDate|null"` The end of the event, or any recurrence of the event, in UTC time must be after this date to match the condition.
- o `*before*`: `"UTCDate|null"` The start of the event, or any recurrence of the event, in UTC time must be before this date to match the condition.
- o `*text*`: `"String|null"` Looks for the text in the `_title_`, `_description_`, `_locations_` (matching name/description), or `_participants_` (matching name/email) properties of the event or any recurrence of the event.
- o `*title*`: `"String|null"` Looks for the text in the `_title_` property of the event, or the overridden `_title_` property of a recurrence.
- o `*description*`: `"String|null"` Looks for the text in the `_description_` property of the event, or the overridden `_description_` property of a recurrence.
- o `*location*`: `"String|null"` Looks for the text in the `_locations_` property of the event (matching name/description of a location), or the overridden `_locations_` property of a recurrence.
- o `*owner*`: `"String|null"` Looks for the text in the name or email fields of a participant in the `_participants_` property of the event, or the overridden `_participants_` property of a recurrence, where the participant has a role of "owner".
- o `*attendee*`: `"String|null"` Looks for the text in the name or email fields of a participant in the `_participants_` property of the event, or the overridden `_participants_` property of a recurrence, where the participant has a role of "attendee".

If zero properties are specified on the `FilterCondition`, the condition MUST always evaluate to `"true"`. If multiple properties are

specified, ALL must apply for the condition to be "true" (it is equivalent to splitting the object into one-property conditions and making them all the child of an AND filter operator).

The exact semantics for matching "String" fields is **deliberately not defined** to allow for flexibility in indexing implementation, subject to the following:

- o Text SHOULD be matched in a case-insensitive manner.
- o Text contained in either (but matched) single or double quotes SHOULD be treated as a **phrase search**, that is a match is required for that exact sequence of words, excluding the surrounding quotation marks. Use "\"", "\"'" and "\"\" to match a literal "\"", "'" and "\" respectively in a phrase.
- o Outside of a phrase, white-space SHOULD be treated as dividing separate tokens that may be searched for separately in the event, but MUST all be present for the event to match the filter.
- o Tokens MAY be matched on a whole-word basis using stemming (so for example a text search for "bus" would match "buses" but not "business").

3.5.2. Sorting

The following properties MUST be supported for sorting:

- o start
- o uid

3.6. CalendarEvent/queryChanges

Standard "/queryChanges" method as described in [I-D.ietf-jmap-core] section 5.6.

4. Security considerations

All security considerations of JMAP ([I-D.ietf-jmap-core]) apply to this specification. Additional considerations specific to the data types and functionality introduced by this document are described in the following subsections.

TODO

5. IANA considerations

5.1. JMAP capability registration for "calendars"

IANA will register the "calendars" JMAP Capability as follows:

Capability Name: "urn:ietf:params:jmap:calendars"

Specification document: this document

Intended use: common

Change Controller: IETF

Security and privacy considerations: this document, section TODO

6. Normative References

[I-D.ietf-calext-jscalendar]

Jenkins, N. and R. Stepanek, "JSCalendar: A JSON representation of calendar data", draft-ietf-calext-jscalendar-12 (work in progress), March 2019.

[I-D.ietf-jmap-core]

Jenkins, N. and C. Newman, "JSON Meta Application Protocol", draft-ietf-jmap-core-17 (work in progress), March 2019.

[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>>.

[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>>.

Authors' Addresses

Neil Jenkins
FastMail
PO Box 234, Collins St West
Melbourne VIC 8007
Australia

Email: neilj@fastmailteam.com
URI: <https://www.fastmail.com>

Michael Douglass
Spherical Cow Group
226 3rd Street
Troy NY 12180
United States of America

Email: mdouglass@sphericalcowgroup.com
URI: <http://sphericalcowgroup.com>