

JMAP
Internet-Draft
Intended status: Standards Track
Expires: September 20, 2018

B. Gondwana, Ed.
FastMail
March 19, 2018

JMAP Extension for imap data
draft-gondwana-jmap-imapdata-00

Abstract

This document adds additional properties to the JMAP Email and Mailbox objects so that servers which also support IMAP can expose metadata about the IMAP Mailstore via 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 September 20, 2018.

Copyright Notice

Copyright (c) 2018 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.

Internet-Draft

JMAP IMAPData

March 2018

Table of Contents

1.	Introduction	2
2.	Conventions Used In This Document	2
3.	Email Object properties	2
3.1.	ImapData Object	3
4.	Mailbox Object properties	3
4.1.	ImapStatus Object	3
5.	Implementation considerations	4
6.	IANA Considerations	4
7.	Security Considerations	4
8.	Acknowledgments	4
9.	Normative References	4
	Author's Address	5

[1.](#) Introduction

[I-D.ietf-jmap-mail] JMAP datastores may be built in such a way that they also allow [\[RFC3501\]](#) IMAP access to the underlying data.

IMAP mailboxes have some STATUS data which is not required for JMAP and hence not exposed by default. This document provides a way to access those values via JMAP.

IMAP mailboxes contain individual messages by UID, and those can have properties which specific to the individual message. If the server supports multiple IMAP messages collapsed into a single JMAP message (due to identical Email/id or [\[I-D.gondwana-imap-uniqueid\]](#) MSGID) then it can be useful to expose the underlying IMAP data via JMAP.

[2.](#) Conventions Used In This Document

In examples, "C:" indicates data sent by a client that is connected to a server. "S:" indicates data sent by the server to the client.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [\[RFC2119\]](#) when they appear in ALL CAPS. These words may also appear in this document in lower case as plain English words, absent their normative meanings.

[3.](#) Email Object properties

This extension adds a single property to the Email object:

- o `*imapdata*`: `String[Integer[ImapData]]` This is a map of mailbox id to a map of uid to ImapData Object

[3.1.](#) ImapData Object

The ImapData Object has the following properties:

- o `*internaldate*`: `Date` This is the date at which the message was created. This may be the same as the JMAP `createdAt` value, or different if the underlying store has different `internaldates` for different messages.
- o `*keywords*`: `String[Boolean]` This is identical to the "keywords" fetch item at the top level, but contains the set of keywords on each individual IMAP message. This may be the same for all messages, depending on the underlying storage mechanics.
- o `*modseq*`: `Integer|null` This is the modseq of the individual message within the IMAP store, or null if the server doesn't support [\[RFC7162\]](#).
- o `*savedate*`: `Date|null` This is the date at which the message was added to this mailbox, or null if the server doesn't support [\[I-D.ietf-extra-imap-savedate\]](#).

[4.](#) Mailbox Object properties

This extension adds a single property to the Mailbox object:

- o `*imapstatus*`: `ImapStatus|null`

If the mailbox is not accessible via IMAP (e.g. a virtual mailbox) then it MUST have a null `ImapStatus`.

[4.1.](#) ImapStatus Object

The ImapStatus Object has the following properties:

- o `*imapname*`: `String` The name of the mailbox in modified UTF7.

"SELECT {imapname}" via IMAP would work if given this name.

- o *highestmodseq*: Integer|null The HIGHESTMODSEQ of the mailbox, or null if the mailbox does not support [[RFC7162](#)].
- o *messages*: Integer The MESSAGES status item (number of messages in mailbox) as defined in [[RFC3501](#)] for the underlying mailbox.
- o *uidvalidity*: Integer The UIDVALIDITY as defined in [[RFC3501](#)] for the underlying mailbox.

Gondwana

Expires September 20, 2018

[Page 3]

Internet-Draft

JMAP IMAPData

March 2018

- o *uidnext*: Integer The UIDNEXT as defined in [[RFC3501](#)] for the underlying mailbox.

[5.](#) Implementation considerations

If the same message occurs multiple times in an IMAP store with different keywords, the combined keyword contents might be best calculated in different ways for different keywords, for example:

- o \$flagged should be set if any IMAP record has \$flagged set
- o \$seen should only be set if ALL messages have \$seen set (because users are generally actually interested in "unseen")

[6.](#) IANA Considerations

There will be a registration of an ID, but there's not yet a JMAP registry to add the it into. Maybe something like "ietf:jmap:imapdata".

[7.](#) Security Considerations

All this data is visible via IMAP already for users with the same authentication rights, however implementations must ensure that if a message is both in mailboxes where the user has the [[RFC4314](#)] READ ACL and other mailboxes where the user does not have read access, that the imapdata response is filtered to avoid leaking information about non-visible mailboxes.

8. Acknowledgments

TBD.

9. Normative References

[I-D.gondwana-imap-uniqueid]

Gondwana, B., "IMAP Extension for unique identifiers", [draft-gondwana-imap-uniqueid-00](#) (work in progress), March 2018.

[I-D.ietf-extra-imap-savedate]

Bosch, S., "Internet Message Access Protocol (IMAP) - SAVEDATE Extension", [draft-ietf-extra-imap-savedate-00](#) (work in progress), March 2018.

[I-D.ietf-jmap-mail]

Jenkins, N., "JMAP for Mail", [draft-ietf-jmap-mail-04](#) (work in progress), March 2018.

Gondwana

Expires September 20, 2018

[Page 4]

Internet-Draft

JMAP IMAPData

March 2018

[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>>.

[RFC3501] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", [RFC 3501](#), DOI 10.17487/RFC3501, March 2003, <<https://www.rfc-editor.org/info/rfc3501>>.

[RFC4314] Melnikov, A., "IMAP4 Access Control List (ACL) Extension", [RFC 4314](#), DOI 10.17487/RFC4314, December 2005, <<https://www.rfc-editor.org/info/rfc4314>>.

[RFC7162] Melnikov, A. and D. Cridland, "IMAP Extensions: Quick Flag Changes Resynchronization (CONDSTORE) and Quick Mailbox Resynchronization (QRESYNC)", [RFC 7162](#), DOI 10.17487/RFC7162, May 2014, <<https://www.rfc-editor.org/info/rfc7162>>.

Author's Address

Bron Gondwana (editor)

FastMail
Level 2, 114 William St
Melbourne VIC 3000
Australia

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