

JMAP

IETF 101 / 22 Mar 2018

Note Well

This is a reminder of IETF policies in effect on various topics such as patents or code of conduct. It is only meant to point you in the right direction. Exceptions may apply. The IETF's patent policy and the definition of an IETF "contribution" and "participation" are set forth in BCP 79; please read it carefully.

As a reminder:

- **By participating in the IETF, you agree to follow IETF processes and policies.**
- **If you are aware that any IETF contribution is covered by patents or patent applications that are owned or controlled by you or your sponsor, you must disclose that fact, or not participate in the discussion.**
- **As a participant in or attendee to any IETF activity you acknowledge that written, audio, video, and photographic records of meetings may be made public.**
- **Personal information that you provide to IETF will be handled in accordance with the IETF Privacy Statement.**
- **As a participant or attendee, you agree to work respectfully with other participants; please contact the ombudsteam (<https://www.ietf.org/contact/ombudsteam/>) if you have questions or concerns about this.**

Definitive information is in the documents listed below and other IETF BCPs. For advice, please talk to WG chairs or ADs:

- **BCP 9 (Internet Standards Process)**
- **BCP 25 (Working Group processes)**
- **BCP 25 (Anti-Harassment Procedures)**
- **BCP 54 (Code of Conduct)**
- **BCP 78 (Copyright)**
- **BCP 79 (Patents, Participation)**

<https://www.ietf.org/privacy-policy/> (Privacy Policy)

Agenda

- Overview of changes since IETF100: 10 min
- JMAP Core remaining issues: 15 min
- JMAP Mail remaining issues: 15 min
- JMAP Extensions/new drafts: 20 min
- Test suite: 20 min
- Other business / GitHub tickets: 10 min

The big rename

- Method naming scheme now Noun/verb
- Response names now match request names
- Result: simpler. No need to pluralise nouns or conjugate verbs. Easier to generate code and harder to f*ck up
- Message renamed to Email to avoid confusion between the JMAP email id, and the RFC5322 Message-ID

Example

```
[[ "Email/get", {  
  "ids": [ "123", "de98a" ],  
  "properties": [ "threadId" ]  
}, "t1" ]]
```



```
[[ "Email/get", {  
  "accountId": "1",  
  "state": "123456",  
  "list": [{  
    "id": "123",  
    "threadId": "trd194",  
  }, {  
    "id": "de98a",  
    "threadId": "trd114"  
  }],  
  "notFound": []  
}, "t1" ]]
```

Sort Collation Algorithms

- Can now specify which collation algorithm to use when sorting the results to a query
- Server may advertise which algorithms it supports
- Collation identifiers from registry established in RFC 4790

Example

```
[[ "Email/query", {  
  "filter": { inMailbox: "1" },  
  "sort": [{  
    "property": "subject",  
    "isAscending": false,  
    "collation": "i;unicode-casemap"  
  }],  
  "position": 0,  
  "limit": 10  
}, "0" ]]
```

RFC5322 JSON representation

- Biggest change since Singapore
- Gives more power to the client without forcing it to parse the message itself
- Simple(ish) option still available

JMAP Core

Remaining Issues

URI for capability

- Each spec (core + mail + future standards/vendor extensions) need an identifier
- Used in JSON session object for server to advertise capabilities it supports, and in the “using” property of a request for clients to opt in to using the spec
- Needs to be unique, preferably short, and ideally help point to documentation.

URI for capability

- [ietf.org/rfc/rfcXXXX.txt](#)
- rfcXXXX + reserve any without a “.” for standards only
- [ietf.org/rfcXXX](#)
- urn:ietf:....:jmap

Proxying

- JMAP is generally very nice for proxying
- A series of method calls may be dispatched to different backends by an intermediate proxy
- But creation ids are currently an issue, as the proxy has to resolve them, which means it has to understand all of the data types so it knows which properties contain ids.

```

[[ "Email/set", {
  "accountId": "primary",
  "create": {
    "k214": {
      "keywords": {
        "$seen": true,
        "$draft": true
      },
      "from": [{
        "name": "John Smith",
        "email": "john@example.com"
      }],
      "to": [{
        "name": "Jane Doe",
        "email": "jane@example.com"
      }],
      "subject": "Example with creation ref",
      ...
    }
  }, "0" ],
...
[ "EmailSubmission/set", {
  "accountId": "primary",
  "create": {
    "k215": {
      "identityId": "100",
      "emailId": "#k214",
      "envelope": null
    }
  }
}, "2" ]]

```

Proxying

- Possible solution: allow an existing set of creation id to actual ids to be passed in, and get back the set with any new ids added.

```
{
  "using": [ ... ],
  "methodCalls": [ ... ],
  "createdIds": {
    "k214": "ae67a9183ffbdca"
  }
}

... response ...

{
  "methodResponses": [ ... ],
  "createdIds": {
    "k214": "ae67a9183ffbdca",
    "k215": "3124eee"
  }
}
```

Anchor offset direction

```
[["Email/query", {  
  "filter": { inMailbox: "1" },  
  "sort": [{ property: "receivedAt" }],  
  "anchor": "msg123",  
  "anchorOffset": 1,  
  "limit": 10  
}, "0" ]]
```

- Portion of results to return determined by position of “anchor” (existing id).
- Should anchorOffset be the offset of the anchor in the results, or the offset of the results from the anchor?

JMAP Mail

Remaining Issues

Inline video/audio

- The textBody/htmlBody lists can include image/* parts
- Should they also be allowed to include video/* and audio/* parts?

Is there a need for a simpler body property?

- The “simple” version is the list of body parts, plus the data in a separate bodyValues property.
- Is there value in an even simpler property, that’s just a HTML string to display as the body.
 - Inline (base64) cid: refs +
 - Upconvert to HTML if only text
 - Concatenate multiple parts if need-be

IANA Considerations

- We need to reference the Mailbox role (SPECIAL-USE) registry, once this has been created (in EXTRA)
- Create a new registry of error types
- Register \$Seen, \$Flagged, \$Answered, \$Draft keywords as reserved for JMAP mapping of IMAP system flags
- Register .well-known
- Anything else?

Overall status

- Both core + mail basically complete
- Will add more examples + IANA considerations, but consider spec semantically done
- Now is really good time to review if you want to make any last minute comments
- Hope to go to last call very soon and publish before IETF 102

**Extensions/
new drafts**

JMAP S/MIME

Alexey Melnikov

JMAP IMAPdata

Bron Gondwana

JMAP Test Suite

Any other business