Intervention: JMAP Portability Motivation

Why?

- Move existing user data between systems over generic API
- Give API spec to legacy systems which have no appropriate API

Example: Migrate away from Horde webmailer.

We present three documents:

1. *Guide*: Bare minimum, even for “one-time migration” use (lower entry barrier)
2. *REST*: Introduce simplified request scheme (even lower barrier)
3. *Extensions*: Solutions to migration-specific problems
JMAP REST Mapping

IETF 119

https://datatracker.ietf.org/doc/draft-ietf-jmap-rest/00/
Changes

It's adopted!
JMAP “REST Mapping”

"apiUrlRest": "https://jmap.me/api/<methodCall>
    ?using=<using>&accountId=<accountId>"

POST /api/ContactCard/get/?
    using=urn%3Aietf%3Aparams%3Ajmap%3Acontacts,
    urn%3Aietf%3Aparams%3Ajmap%3Acore&accountId=U7339402f
Host: jmap.me
Accept: application/json

Neil: “odd”

Mark: “it has more to do with RPC than REST”
JMAP "REST" Mapping

Low requirements for low effort JMAP implementation:

- No JSON in Request
- No Routing in URI
JMAP "REST" Mapping

Low requirements for low effort JMAP implementation:

- No JSON in Request
- No Routing in URI

Neil: “Do only CRUD, but do:”

```javascript
apiRestUrl: `/api/{accountId}/{typeName}
GET /api/account1/Calendar/123-12345?properties=["name", "color"]
```
JMAP "REST" Mapping

Low requirements for low effort JMAP implementation:

- No JSON in Request
- No Routing in URI

Neil: “Do only CRUD, but do:”

```javascript
apiRestUrl: "/api/{accountId}/{typeName}
GET /api/account1/Calendar/123-12345?properties=["name", "color"]
```

Me:

- Seems fine
- Is /query part of simple CRUD? Would be good to leave out for now.
URI Templates RFC6570

"apiUrlRest": "https://jmap.me/api/<methodCall>
    ?using=<using>&accountId=<accountId>"

VS

"apiUrlRest": "https://jmap.me/api/{methodCall}
    ?using={using}&accountId={accountId}"

- For now sticking close to Session.downloadUrl of RFC8620 (JMAP Core)
- Could move to URI template
- If yes, do we want to allow any kind of URI template as a value?