

# IMAP PARTIAL (paged Search and Fetch) extension

draft-melnikov-imap-partial-00.txt  
Alexey Melnikov <[alexey.melnikov@isode.com](mailto:alexey.melnikov@isode.com)>  
and Yahoo team

# Problem Statement

- How to handle large IMAP mailboxes in an efficient way?
- How “large” is “large”?
  - 50k+ messages
- What is exactly the problem?
  - Memory/resource usage of keeping msgno-to-UID map on the server
  - Some clients can't even provide user access to that many messages
  - Clients blindly doing [UID] FETCH 1:\* FLAGS every so often, even if there are no changes
    - CONDSTORE/QRESYNC are your friends, but more can be done

# Proposed PARTIAL extension

- New SEARCH return option for returning a "page" of search results at a time
  - This can reduce the amount of work a server has to do, as the server can stop processing SEARCH once the requested page is filled in
  - Also reduces the amount of data sent by the server over IMAP
- Similarly, a new UID FETCH modifier that can restrict the number of messages processed

# Proposed PARTIAL extension: details (1 of 2)

- Paged SEARCH/UID SEARCH using PARTIAL search result option
  - Originally defined in RFC 5267 (part of CONTEXT=SEARCH)
  - Extended to allow Python-like negative ranges (e.g. "-1:-3" -- "the last 3" resulting messages)

Example:

```
A01 UID SEARCH RETURN (PARTIAL -1:-100)  
UNDELETED UNKEYWORD $Junk
```

# Proposed PARTIAL extension: details (2 of 2)

- Paged UID FETCH using PARTIAL FETCH modifier
  - Same syntax as for the SEARCH
  - Can be used to limit the number of messages in a UID set, when the number of messages is not known

Example:

10 UID FETCH ***25900:26600*** (UID FLAGS) (**PARTIAL -1:-15**)

# Bonus Features

- The draft clarifies interaction between PARTIAL and SAVE return options

# Next steps

- Ask for adoption in the EXTRA WG