

IMAP Extension: Status-Counters

**(draft-neystadt-imap-
status-counters-01.txt)**

IETF-54 Update

John Neystadt

Alexey Melnikov

Ari Erev

July 15, 2002

IMAP: Status-Counters

- In a Multimedia / Multi-Context the message store contains messages of different “Contexts”
(internet-draft (draft-ietf-vpim-hint-08.txt))
- The draft implements a requirement from um-issues internet draft (draft-vaudreuil-um-issues-00.txt) to extend the “Mailbox Summary” to have per-context counters.
- Provides aggregate information based on message types/contexts.
- Basically a performance/optimization to save on network and CPU resources.

Status-Counters Overview

- Extends the IMAP STATUS command.
- A new meta-data item – ‘COUNTERS’
 - In addition to: ‘Recent’, ‘Messages’, ‘UIDNEXT’
- Expresses a list of counter-names (or attributes).
- Provides the client with different numeric summaries of messages - grouped according to the value of Message-Context header field.
- Published in ‘Capability’ as: “STATUS-COUNTERS”.

Status-Counters – use case

- The main use case is:
- Typical voice-mail sessions starts:
"you have 2 new voice messages, 6 unheard messages, and 1 new fax message".
- Currently - this information can be calculated using:
 - Option 1:
 - | For a given mailbox, the client issues:
 - | a SELECT mailbox
 - | b FETCH 1:* (BODY[HEADER (Message-Context)] FLAGS)
 - | Calculate the counters from FETCH responses

Or:

Status-Counters – use case (cont.)

I Option 2:

 a SELECT mailbox

 b SEARCH HEADER "Message-Context" "Voice-Message"


Construct message set from the result; if is empty, skip till step 5.

 c SEARCH UNSEEN.

Construct message set from the result (is the list of all unseen voice messages); if is empty, skip till step 5.

 d SEARCH KEYWORD \$Important .

Result is the list of all important unseen voice messages;

 Repeat steps 2-4 for each message class the client is interested in.

 Calculate the counters from SEARCH responses

Status-Counters Advantage

- The proposed extension suggest to optimize performance (Network as well as CPU resources) by standardizing a set of known “Queries”.
- Per-context counting is done on the server and only results are transferred to the client.
- Much simpler for the client.
- Possibly less resources on the server (which can cache/ or pre-compute the counters).

Status-Counters example

■ C: A CAPABILITY
S: * CAPABILITY ... IMAP4rev1 STATUS-COUNTERS ... S: S: A
OK CAPABILITY
...
C: B STATUS Inbox (UIDNEXT UIDVALIDITY /
COUNTERS (ÄSeen \$Important "Unseen-Important"
ÄRecent))
S: * STATUS Inbox (UIDNEXT 850 UIDVALIDITY 1234
COUNTERS
(All (100 ÄSeen 30 \$Important 20
"Unseen-Important" 10 ÄRecent 5)
"Voice-Message" (10 ÄSeen 3 \$Important 2
"Unseen-Important" 1 ÄRecent 2)
"Fax-Message" (10 ÄSeen 3 \$Important 2
"Unseen-Important" 1 ÄRecent 3)))
S: B OK STATUS completed.

Status-Counters: \$Important

- STATUS-COUNTERS relies on a special support of \$Important keyword defined in [IMAP-KEYWORDS] by the IMAP server.
- The server MUST automatically set \$Important flag on injection of an "important" message as described in IMAP-KEYWORDS I-d (draft-melnikov-imap-keywords-00.txt]. Basically:
 - If the root body-part of that message contains the header field "Importance" with the value "High".
 - If the root body-part contains either a header field "Priority" with the value of "urgent", or a header field "X-Priority" with the value "1" or "2".

For Further Information...

- UM-LEMONADE mailing list

- send 'subscribe um'

- to:

- Majordomo@mail.snowshore.com

- Contact us

- John Neystadt (john@comverse.com)

- Alexey Melnikov
(mel@messagingdirect.com)