

Network Working Group
Internet Draft
Document: [draft-segmuller-imap-structure-00.txt](#)

W. Segmuller
IBM T.J. Watson Research Center
April 2001
Expires: October 2001

IMAP Extension: Structure

Status of this Document

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#). Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at
<http://www.ietf.org/ietf/lid-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at
<http://www.ietf.org/shadow.html>.

The protocol discussed in this document is experimental and subject to change. Persons planning on either implementing or using this protocol are STRONGLY URGED to get in touch with the author before embarking on such a project.

Discussion and suggestions for improvement are requested, and should be sent to ietf-imapext@imc.org. This document will expire before 31 October 2001. Distribution of this memo is unlimited.

Abstract

This document describes the STRUCTURE extension to the [IMAP] protocol. This extension allows the querying of the message structure information in a self describing way.

[1](#) Introduction

This document describes a new static message data item which returns all the content fields of all the parts in a message.

Clients that implement this extension need only issue one command to obtain all the content-* fields (defined in [MIME]) of all part.

Internet Draft

IMAP Extension: Structure

April 2001

Without this extension, this is not possible if there is a required content-* field which is not in the BODYSTRUCTURE command. A client must wait for the fetch BODYSTRUCTURE to return, and then issue fetch "BODY[n.MIME]" for each part that it requires this additional information for.

We can add these fields to the BODYSTRUCTURE response, but then the problem will hit us again in the future.

If there are two extensions that add fields to the BODYSTRUCTURE response at the same time, then these extensions need to be aware of each other when being defined, otherwise we will have two conflicting extensions.

Draft Note:

This problem recently surfaced in the "Voice Profile for Internet Mail" IETF working group [VPIM]. One part of the discussion centered on where to put the critical content notation such that an IMAP client can get the information without an extra round trip. This caused them to look at where the critical content can be squeezed into instead of what is right for MIME.

[VPIM]; Discussions on "Voice Profile for Internet Mail" IETF working group; Charter:
"http://www.ietf.org/html.charters/vpim-charter.html"; Archive:
"http://www.neystadt.org/vpim/"; March 2001

2 Conventions used in this document

Conventions for notations are as in [IMAP], including the use of [KEYWORDS].

The capability string associated with the extension defined in this document is "STRUCTURE".

3 FETCH STRUCTURE Command

The STRUCTURE message data item is added to the fetch command. This static data item returns the structure of the message. The data returned MUST NOT change for any particular message.

For each part in the message, the "content-" fields are returned. The FIELDS and FIELDS.NOT part specifiers, which are followed by a

list of field-name (as defined in [[RFC822](#)]) names, may be used to restrict which fields are returned. The content-type and content-disposition fields are always returned. The FIELDS part specifier return the specified fields, in addition to the content-type and content-disposition fields. The FIELDS.NOT part specifier returns only the "content-" fields which are not listed, but always returns the content-type and content-disposition.

[4](#) FETCH STRUCTURE Response

The STRUCTURE message data item returns information about all the parts in a message. For every part in the message, the content-type (with parameters), the content-disposition (with parameters), and all the "content-" fields are returned. The content-* fields that are returned are determined by the FIELDS and FIELDS.NOT specifier of the FETCH STRUCTURE command. The content-type and content-disposition are always returned.

Following the content-* fields, some additional information, which is dependent on the type of part, is returned.

For "text/*" parts, the number of octets and the number of lines in the body of the part is returned.

For "multipart/*" parts, the STRUCTURE of each nested part is returned.

For "message/rfc822" parts, the ENVELOPE and STRUCTURE of the nested message is returned.

For all other parts, the number of octets in the body of the part is returned.

[5](#) Formal Syntax

This syntax updates the formal syntax in [IMAP].

The following syntax specification uses the Augmented Backus-Naur Form (ABNF) notation as specified in [ABNF].

```
fetch-att =/ "STRUCTURE" [ "[" "FIELDS" [ ".NOT" ] SP header-list
```

"]"]

msg-att-static =/ "STRUCTURE" SP structure

```
structure = "(" media-content-type
             SP body-fld-param
             SP body-fld-dsp
             SP structure-content-fields
             SP structure-body
             ")"
```

media-content-type = media-type SP media-subtype

media-type = string

```
structure-content-fields = "(" structure-field
                           *(SP structure-field)
                           ")" / nil
                           ; all content-* fields in the
                           ; part header
                           ; except content-type and
                           ; content-disposition
```

structure-field = structure-header SP structure-value

structure-header = string

structure-value = string

structure-body = structure-body-basic / structure-body-text
/ structure-body-msg / structure-body-mpart

structure-body-basic = body-fld-octets
; content-type of anything except
; "text/*", "multipart/*",

"message/rfc822"

```

structure-body-text = body-fld-octets SPACE body-fld-lines
                      ; content-type of "text/*"

structure-body-msg = body-fld-octets SP body-fld-lines
                     SP envelope SP structure
                     ; content-type of "message/rfc822"

structure-body-mpart = structure *( SP structure )
                      ; content-type of "multipart/*"

```

Draft Note:

The following are copied from [IMAP], they are not being defined in this document.

```

body-fld-dsp      = "(" string SP body-fld-param ")" / nil

body-fld-lines   = number

body-fld-octets  = number

body-fld-param   = "(" string SP string *(SP string SP string)
                  ")" / nil

envelope         = "(" env-date SP env-subject
                  SP env-from SP env-sender SP env-reply-to
                  SP env-to SP env-cc SP env-bcc
                  SP env-in-reply-to SP env-message-id ")"

media-subtype    = string

```

[6](#) Security Considerations

Security considerations are discussed in [IMAP].

It is belived that this extension doesn't introduce any additional security concerns.

[7](#) Example

For the message:

```
to: foo@bar.com.invalid
from: bar@foo.com.invalid
subject: re: foo bar baz
content-type: multipart/mixed; boundary="A"

--A
content-type: text/html; charset="iso-8859-1"
content-transfer-encoding: quoted-printable

hello

--A
content-type: image/gif; name="xyz.gif"
content-transfer-encoding: base64
content-disposition: attachment; filename="xyz.gif"

....

--A
content-type: message/rfc822

from: foo@bar.com.invalid
to: bar@foo.com.invalid
subject: foo bar baz
content-type: text/plain

hello

--A--
```

The command:

```
xx FETCH 1 (STRUCTURE)
```

Would return:

```
* 1 FETCH (STRUCTURE
    ("multipart" "mixed" ("boundary" "A") nil nil
      ("text" "html" ("charset" "iso-8859-1") nil
        ("content-transfer-encoding" "quoted-printable")
        7 1 )
      ("image" "gif" ("name" "xyz.gif")
        ("attachment" ("filename" "xyz.gif"))
        ("content-transfer-encoding" "quoted-printable")
        200 )
      ("message" "rfc822" nil nil nil
        200 7
        <envelope information for nested message>
        ("text" "html" ("charset" "iso-8859-1") nil
          nil 7 1 )
      )
    )
  )
```

In the above example, the one long line has been folded, and extra spaces have been inserted at the beginning of each line for formatting purposes only. In the actual response, there is only one space between tokens, and no embedded new line characters.

[8](#) Author's Address

Wolfgang Segmuller
IBM T.J. Watson Research Center
30 Saw Mill River Rd
Hawthorne, NY 10532

Phone: 1-914-784-7408
Email: whs@watson.ibm.com

[Appendix A](#). References

[IMAP]; Crispin, M.; "Internet Message Access Protocol - Version 4rev1"; Internet Draft [draft-crispin-imapv-14.txt](#) (updates [RFC 2060](#)); University of Washington; January 2001

[KEYWORDS]; Bradner, S.; "Key words for use in RFCs to Indicate Requirement Levels"; [RFC 2119](#); Harvard University; March 1997

[ABNF]; Crocker, D.; "Augmented BNF for Syntax Specifications: ABNF"; [RFC 2234](#); Internet Mail Consortium; November, 1997.

[[RFC822](#)]; Crocker, D.; "Standard for the Format of ARPA Internet Text Messages"; STD 11; [RFC 822](#); University of Delaware; August 1982

[MIME]; Freed, N. and Borenstein, N.; "Multipurpose Internet Mail

Extensions (MIME) Part One: Format of Internet Message Bodies", [RFC 2045](#); Innosoft and First Virtual; November 1996.

Segmuller

Expires October 2001

[Page 6]

Internet Draft

IMAP Extension: Structure

April 2001

[Appendix B](#). Full Copyright Statement

Copyright (C) The Internet Society 2001. All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

