

## The Text/Plain Format Parameter

### Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

### Copyright Notice

Copyright (C) The Internet Society (1999). All Rights Reserved.

### Table of Contents

|  |    |
|--|----|
| 1. Abstract . . . . .  | 2  |
| 2. Conventions Used in this Document . . . . .                 | 2  |
| 3. The Problem . . . . .                                       | 2  |
| 3.1. Paragraph Text . . . . .                                  | 3  |
| 3.2. Embarrassing Line Wrap . . . . .                          | 3  |
| 3.3. New Media Types . . . . .                                 | 4  |
| 4. The Format Parameter to the Text/Plain Media Type . . . . . | 4  |
| 4.1. Generating Format=Flowed . . . . .                        | 5  |
| 4.2. Interpreting Format=Flowed . . . . .                      | 6  |
| 4.3. Usenet Signature Convention . . . . .                     | 7  |
| 4.4. Space-Stuffing . . . . .                                  | 7  |
| 4.5. Quoting . . . . .   | 8  |
| 4.6. Digital Signatures and Encryption . . . . .               | 9  |
| 4.7. Line Analysis Table . . . . .                             | 10 |
| 4.8. Examples . . . . .  | 10 |
| 5. ABNF . . . . .  | 11 |
| 6. Failure Modes . . . . .                                     | 11 |
| 6.1. Trailing White Space Corruption . . . . .                 | 11 |
| 7. Security Considerations . . . . .                           | 12 |
| 8. IANA Considerations . . . . .                               | 12 |
| 9. Internationalization Considerations . . . . .               | 12 |
| 10. Acknowledgments . . . . .                                  | 12 |
| 11. References . . . . .                                       | 13 |
| 12. Editor's Address . . . . .                                 | 13 |
| 13. Full Copyright Statement . . . . .                         | 14 |

## 1. Abstract

Interoperability problems have been observed with erroneous labelling of paragraph text as Text/Plain, and with various forms of "embarrassing line wrap." (See section 3.)

Attempts to deploy new media types, such as Text/Enriched [RICH] and Text/HTML [HTML] have suffered from a lack of backwards compatibility and an often hostile user reaction at the receiving end.

What is required is a format which is in all significant ways Text/Plain, and therefore is quite suitable for display as Text/Plain, and yet allows the sender to express to the receiver which lines can be considered a logical paragraph, and thus flowed (wrapped and joined) as appropriate.

This memo proposes a new parameter to be used with Text/Plain, and, in the presence of this parameter, the use of trailing whitespace to indicate flowed lines. This results in an encoding which appears as normal Text/Plain in older implementations, since it is in fact normal Text/Plain.

## 2. Conventions Used in this Document

The key words "REQUIRED", "MUST", "MUST NOT", "SHOULD", "SHOULD NOT", and "MAY" in this document are to be interpreted as described in "Key words for use in RFCs to Indicate Requirement Levels" [KEYWORDS].

## 3. The Problem

The Text/Plain media type is the lowest common denominator of Internet email, with lines of no more than 997 characters (by convention usually no more than 80), and where the CRLF sequence represents a line break [MIME-INT].

Text/Plain is usually displayed as preformatted text, often in a fixed font. That is, the characters start at the left margin of the display window, and advance to the right until a CRLF sequence is seen, at which point a new line is started, again at the left margin. When a line length exceeds the display window, some clients will wrap the line, while others invoke a horizontal scroll bar.

Text which meets this description is defined by this memo as "fixed".

Some interoperability problems have been observed with this media type:

### 3.1. Paragraph Text

Many modern programs use a proportional-spaced font and CRLF to represent paragraph breaks. Line breaks are "soft", occurring as needed on display. That is, characters are grouped into a paragraph until a CRLF sequence is seen, at which point a new paragraph is started. Each paragraph is displayed, starting at the left margin (or paragraph indent), and continuing to the right until a word is encountered which does not fit in the remaining display width. This word is displayed at the left margin of the next line. This continues until the paragraph ends (a CRLF is seen). Extra vertical space is left between paragraphs.

Text which meets this description is defined by this memo as "flowed".

Numerous software products erroneously label this media type as Text/Plain, resulting in much user discomfort.

### 3.2. Embarrassing Line Wrap

As Text/Plain messages get quoted in replies or forwarded messages, the length of each line gradually increases, resulting in "embarrassing line wrap." This results in text which is at best hard to read, and often confuses attributions.

Example:

```
>>>>>This is a comment from the first message to show a
>quoting example.
>>>>>This is a comment from the second message to show a
>quoting example.
>>>>This is a comment from the third message.
>>>This is a comment from the fourth message.
```

It can be confusing to assign attribution to lines 2 and 4 above.

In addition, as devices with display widths smaller than 80 characters become more popular, embarrassing line wrap has become even more prevalent, even with unquoted text.

Example:

This is paragraph text that is meant to be flowed across several lines. However, the sending mailer is converting it to fixed text at a width of 72 characters, which causes it to look like this when shown on a PDA with only 30 character lines.

### 3.3. New Media Types

Attempts to deploy new media types, such as Text/Enriched [RICH] and Text/HTML [HTML] have suffered from a lack of backwards compatibility and an often hostile user reaction at the receiving end.

In particular, Text/Enriched requires that open angle brackets (" $<$ ") and hard line breaks be doubled, with resulting user unhappiness when viewed as Text/Plain. Text/HTML requires even more alteration of text, with a corresponding increase in user complaints.

A proposal to define a new media type to explicitly represent the paragraph form suffered from a lack of interoperability with currently deployed software. Some programs treat unknown subtypes of Text as an attachment.

What is desired is a format which is in all significant ways Text/Plain, and therefore is quite suitable for display as Text/Plain, and yet allows the sender to express to the receiver which lines can be considered a logical paragraph, and thus flowed (wrapped and joined) as appropriate.

## 4. The Format Parameter to the Text/Plain Media Type

This document defines a new MIME parameter for use with Text/Plain:

Name: Format  
Value: Fixed, Flowed

(Neither the parameter name nor its value are case sensitive.)

If not specified, a value of Fixed is assumed. The semantics of the Fixed value are the usual associated with Text/Plain [MIME-INT].

A value of Flowed indicates that the definition of flowed text (as specified in this memo) was used on generation, and MAY be used on reception.

This section discusses flowed text; section 5 provides a formal definition.

Because flowed lines are all-but-indistinguishable from fixed lines, currently deployed software treats flowed lines as normal Text/Plain (which is what they are). Thus, no interoperability problems are expected.

Note that this memo describes an on-the-wire format. It does not address formats for local file storage.

#### 4.1. Generating Format=Flowed

When generating Format=Flowed text, lines SHOULD be shorter than 80 characters. As suggested values, any paragraph longer than 79 characters in total length could be wrapped using lines of 72 or fewer characters. While the specific line length used is a matter of aesthetics and preference, longer lines are more likely to require rewrapping and to encounter difficulties with older mailers. It has been suggested that 66 character lines are the most readable.

(The reason for the restriction to 79 or fewer characters between CRLFs on the wire is to ensure that all lines, even when displayed by a non-flowed-aware program, will fit in a standard 80-column screen without having to be wrapped. The limit is 79, not 80, because while 80 fit on a line, the last column is often reserved for a line-wrap indicator.)

When creating flowed text, the generating agent wraps, that is, inserts 'soft' line breaks as needed. Soft line breaks are added between words. Because a soft line break is a SP CRLF sequence, the generating agent creates one by inserting a CRLF after the occurrence of a space.

A generating agent SHOULD NOT insert white space into a word (a sequence of printable characters not containing spaces). If faced with a word which exceeds 79 characters (but less than 998 characters, the [SMTP] limit on line length), the agent SHOULD send the word as is and exceed the 79-character limit on line length.

A generating agent SHOULD:

1. Ensure all lines (fixed and flowed) are 79 characters or fewer in length, counting the trailing space but not counting the CRLF, unless a word by itself exceeds 79 characters.
2. Trim spaces before user-inserted hard line breaks.
3. Space-stuff lines which start with a space, "From ", or ">".

In order to create messages which do not require space-stuffing, and are thus more aesthetically pleasing when viewed as Format=Fixed, a generating agent MAY avoid wrapping immediately before ">", "From ", or space.

(See sections 4.4 and 4.5 for more information on space-stuffing and quoting, respectively.)

A Format=Flowed message consists of zero or more paragraphs, each containing one or more flowed lines followed by one fixed line. The usual case is a series of flowed text lines with blank (empty) fixed lines between them.

Any number of fixed lines can appear between paragraphs.

[Quoted-Printable] encoding SHOULD NOT be used with Format=Flowed unless absolutely necessary (for example, non-US-ASCII (8-bit) characters over a strictly 7-bit transport such as unextended SMTP). In particular, a message SHOULD NOT be encoded in Quoted-Printable for the sole purpose of protecting the trailing space on flowed lines unless the body part is cryptographically signed or encrypted (see Section 4.6).

The intent of Format=Flowed is to allow user agents to generate flowed text which is non-obnoxious when viewed as pure, raw Text/Plain (without any decoding); use of Quoted-Printable hinders this and may cause Format=Flowed to be rejected by end users.

#### 4.2. Interpreting Format=Flowed

If the first character of a line is a quote mark (">"), the line is considered to be quoted (see section 4.5). Logically, all quote marks are counted and deleted, resulting in a line with a non-zero quote depth, and content. (The agent is of course free to display the content with quote marks or excerpt bars or anything else.) Logically, this test for quoted lines is done before any other tests (that is, before checking for space-stuffed and flowed).

If the first character of a line is a space, the line has been space-stuffed (see section 4.4). Logically, this leading space is deleted before examining the line further (that is, before checking for flowed).

If the line ends in one or more spaces, the line is flowed. Otherwise it is fixed. Trailing spaces are part of the line's content, but the CRLF of a soft line break is not.

A series of one or more flowed lines followed by one fixed line is considered a paragraph, and MAY be flowed (wrapped and unwrapped) as appropriate on display and in the construction of new messages (see section 4.5).

A line consisting of one or more spaces (after deleting a stuffed space) is considered a flowed line.

#### 4.3. Usenet Signature Convention

There is a convention in Usenet news of using "-- " as the separator line between the body and the signature of a message. When generating a Format=Flowed message containing a Usenet-style separator before the signature, the separator line is sent as-is. This is a special case; an (optionally quoted) line consisting of DASH DASH SP is not considered flowed.

#### 4.4. Space-Stuffing

In order to allow for unquoted lines which start with ">", and to protect against systems which "From-munge" in-transit messages (modifying any line which starts with "From " to ">From "), Format=Flowed provides for space-stuffing.

Space-stuffing adds a single space to the start of any line which needs protection when the message is generated. On reception, if the first character of a line is a space, it is logically deleted. This occurs after the test for a quoted line, and before the test for a flowed line.

On generation, any unquoted lines which start with ">", and any lines which start with a space or "From " SHOULD be space-stuffed. Other lines MAY be space-stuffed as desired.

(Note that space-stuffing is similar to dot-stuffing as specified in [SMTP].)

If a space-stuffed message is received by an agent which handles Format=Flowed, the space-stuffing is reversed and thus the message appears unchanged. An agent which is not aware of Format=Flowed will of course not undo any space-stuffing, thus Format=Flowed messages may appear with a leading space on some lines (those which start with a space, ">" which is not a quote indicator, or "From "). Since lines which require space-stuffing rarely occur, and the aesthetic consequences of unreversed space-stuffing are minimal, this is not expected to be a significant problem.

#### 4.5. Quoting

In Format=Flowed, the canonical quote indicator (or quote mark) is one or more close angle bracket (">") characters. Lines which start with the quote indicator are considered quoted. The number of ">" characters at the start of the line specifies the quote depth. Flowed lines which are also quoted may require special handling on display and when copied to new messages.

When creating quoted flowed lines, each such line starts with the quote indicator.

Note that because of space-stuffing, the lines

```
>> Exit, Stage Left
```

and

```
>>Exit, Stage Left
```

are semantically identical; both have a quote-depth of two, and a content of "Exit, Stage Left".

However, the line

```
> > Exit, Stage Left
```

is different. It has a quote-depth of one, and a content of "> Exit, Stage Left".

When generating quoted flowed lines, an agent needs to pay attention to changes in quote depth. A sequence of quoted lines of the same quote depth SHOULD be encoded as a paragraph, with the last line generated as fixed and prior lines generated as flowed.

If a receiving agent wishes to reformat flowed quoted lines (joining and/or wrapping them) on display or when generating new messages, the lines SHOULD be de-quoted, reformatted, and then re-quoted. To de-quote, the number of close angle brackets in the quote indicator at the start of each line is counted. Consecutive lines with the same quoting depth are considered one paragraph and are reformatted together. To re-quote after reformatting, a quote indicator containing the same number of close angle brackets originally present is prefixed to each line.



On reception, if a change in quoting depth occurs on a flowed line, this is an improperly formatted message. The receiver SHOULD handle this error by using the 'quote-depth-wins' rule, which is to ignore the flowed indicator and treat the line as fixed. That is, the change in quote depth ends the paragraph.

For example, consider the following sequence of lines (using '\*' to indicate a soft line break, i.e., SP CRLF, and '#' to indicate a hard line break, i.e., CRLF):

```
> Thou villainous ill-breeding spongy dizzy-eyed*
> reeky elf-skinned pigeon-egg!*      <--- problem ---<
>> Thou artless swag-bellied milk-livered*
>> dismal-dreaming idle-headed scut!#
>>> Thou errant folly-fallen spleeny reeling-ripe*
>>> unmuzzled ratsbane!#
>>>> Henceforth, the coding style is to be strictly*
>>>> enforced, including the use of only upper case.#
>>>>> I've noticed a lack of adherence to the coding*
>>>>> styles, of late.#
>>>>> Any complaints?#
```

The second line ends in a soft line break, even though it is the last line of the one-deep quote block. The question then arises as to how this line should be interpreted, considering that the next line is the first line of the two-deep quote block.

The example text above, when processed according to quote-depth wins, results in the first two lines being considered as one quoted, flowed section, with a quote depth of 1; the third and fourth lines become a quoted, flowed section, with a quote depth of 2.

A generating agent SHOULD NOT create this situation; a receiving agent SHOULD handle it using quote-depth wins.

#### 4.6. Digital Signatures and Encryption

If a message is digitally signed or encrypted it is important that cryptographic processing use the on-the-wire Format=Flowed format. That is, during generation the message SHOULD be prepared for transmission, including addition of soft line breaks, space-stuffing, and [Quoted-Printable] encoding (to protect soft line breaks) before being digitally signed or encrypted; similarly, on receipt the message SHOULD have the signature verified or be decrypted before [Quoted-Printable] decoding and removal of stuffed spaces, soft line breaks and quote marks, and reflowing.

#### 4.7. Line Analysis Table

Lines contained in a Text/Plain body part with Format=Flowed can be analyzed by examining the start and end of the line. If the line starts with the quote indicator, it is quoted. If the line ends with one or more space characters, it is flowed. This is summarized by the following table:

| Starts<br>with<br>Quote<br>----- | Ends in<br>One or<br>More Spaces<br>----- | Line<br>Type<br>----- |
|----------------------------------|---|-----------------------|
| no                               | no  | unquoted, fixed       |
| yes                              | no  | quoted, fixed         |
| no                               | yes                                       | unquoted, flowed      |
| yes                              | yes                                       | quoted, flowed        |

#### 4.8. Examples

The following example contains three paragraphs:

'Take some more tea,' the March Hare said to Alice, very earnestly.

'I've had nothing yet,' Alice replied in an offended tone, 'so I can't take more.'

'You mean you can't take LESS,' said the Hatter: 'it's very easy to take MORE than nothing.'

This could be encoded as follows (using '\*' to indicate a soft line break, that is, SP CRLF sequence, and '#' to indicate a hard line break, that is, CRLF):

```
'Take some more tea,' the March Hare said to Alice, very*
earnestly.*
#
'I've had nothing yet,' Alice replied in an offended tone, 'so*
I can't take more.'*
#
'You mean you can't take LESS,' said the Hatter: 'it's very*
easy to take MORE than nothing.'#
```

To show an example of quoting, here we have the same exchange, presented as a series of direct quotes:

```
>>>Take some more tea.#
>>I've had nothing yet, so I can't take more.#
>You mean you can't take LESS, it's very easy to take*
>MORE than nothing.#
```

## 5. ABNF

The constructs used in Text/Plain; Format=Flowed body parts are described using [ABNF], including the Core Rules:

```
paragraph      = 1*flowed-line fixed-line
fixed-line     = fixed / sig-sep
fixed          = [quote] [stuffing] *text-char non-sp CRLF
flowed-line    = flow-qt / flow-unqt
flow-qt        = quote [stuffing] *text-char 1*SP CRLF
flow-unqt      = [stuffing] *text-char 1*SP CRLF
non-sp         = %x01-09 / %x0B / %x0C / %x0E-1F / %x21-7F
                ; any 7-bit US-ASCII character, excluding
                ; NUL, CR, LF, and SP

quote          = 1*">"
sig-sep        = [quote] "--" SP CRLF
stuffing       = [SP] ; space-stuffed, added on generation if
                ; needed, deleted on reception
text-char      = non-sp / SP
```

## 6. Failure Modes

### 6.1. Trailing White Space Corruption

There are systems in existence which alter trailing whitespace on messages which pass through them. Such systems may strip, or in rarer cases, add trailing whitespace, in violation of RFC 821 [SMTP] section 4.5.2.

Stripping trailing whitespace has the effect of converting flowed lines to fixed lines, which results in a message no worse than if Format=Flowed had not been used.

Adding trailing whitespace to a Format=Flowed message may result in a malformed display or reply.

Since most systems which add trailing white space do so to create a line which fills an internal record format, the result is almost always a line which contains an even number of characters (counting the added trailing white space).

One possible avoidance, therefore, would be to define Format=Flowed lines to use either one or two trailing space characters to indicate a flowed line, such that the total line length is odd. However, considering the scarcity of such systems today, it is not worth the added complexity.

## 7. Security Considerations

This parameter introduces no security considerations beyond those which apply to Text/Plain.

Section 4.6 discusses the interaction between Format=Flowed and digital signatures or encryption.

## 8. IANA Considerations

IANA is requested to add a reference to this specification in the Text/Plain Media Type registration.

## 9. Internationalization Considerations

The line wrap and quoting specifications of Format=Flowed may not be suitable for certain charsets, such as for Arabic and Hebrew characters that read from right to left. Care should be taken in applying format=flowed in these cases, as format=fixed combined with quoted-printable encoding may be more suitable.

## 10. Acknowledgments

This proposal evolved from a discussion of Chris Newman's Text/Paragraph draft which took place on the IETF 822 mailing list. Special thanks to Ian Bell, Steve Dorner, Brian Kelley, Dan Kohn, Laurence Lundblade, and Dan Wing for their reviews, comments, suggestions, and discussions.

## 11. References

- |            |   |
|------------|---|
| [ABNF]     | Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", RFC 2234, November 1997. |
| [KEYWORDS] | S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997. |
| [RICH]     | Resnick, P. and A. Walker, "The text/enriched MIME Content-type", RFC 1896, February 1996.            |

- [MIME-IMT] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types", RFC 2046, November 1996.
- [Quoted-Printable] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", RFC 2045, November 1996.
- [SMTP] Postel, J., "Simple Mail Transfer Protocol", STD 10, RFC 821, August 1982.
- [HTML] Berners-Lee, T. and D. Connolly, "Hypertext Markup Language -- 2.0", RFC 1866, November 1995.

## 12. Editor's Address

Randall Gellens  
QUALCOMM Incorporated  
5775 Morehouse Dr.  
San Diego, CA 92121-2779  
USA

Phone: +1 619 651 5115  
EMail: randy@qualcomm.com

### 13. Full Copyright Statement

Copyright (C) The Internet Society (1999). 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.

### Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.

