付録 :RFC1522 日本語訳
訳 :藤本真吾 (shingo@wide.ad.jp)

Network Working Group
Request for Comments: 1522
Obsoletes: 1342
Category: Standards Track

K.Moore
University of Tennessee
September 1993

MIME (Multipurpose Internet Mail Extensions) パート II
非 ASCII テキスト対応メッセージヘッダの拡張

メモのステータス

この RFC では、インターネット社会におけるインターネット ・スタンダード ・トラックとなったプロトコルの仕様を定義します。また、よりよい発展のために、このプロトコルについての議論、提案を広く求めます。このプロトコルの現在の規格化のステート、およびステータスの詳細については、最新版の 『Internet Official Protocol Standards』を参照してください。このメモの再配布は、自由におこなってください。

概要

このメモでは、RFC822 (STD 11) でメッセージヘッダとして定められた非 ASCII の文字セットを使った表現を許す、RFC1521[1] で定義されたメッセージフォーマットへの拡張について説明します。ここで説明するヘッダへの拡張は、現在インターネットで広く使われているメールのハンドリングをするソフトウェアとの共存ができるように考慮されています。またこの拡張により、RFC1521 形式のメールリーダの実装を容易にすることを狙っています。

1 はじめに

RFC1521 では、様々な文字セットで記述されているテキストのボディパートを、表示可能な ASCII コード文字へとエンコードすることでその表現を可能にする仕組みについて説明しました。このメモでは、非 ASCII コードのテキストを既存のメッセージハンドラを混乱させないようにエンコードして、RFC822[2] 準拠のメッセージヘッダの一部として使えるようにする方法について説明します。

RFC1521 で説明されているエンコードの場合と同様に、この手法では既存のインターネットで使われているメールハンドルプログラムを混乱させることなく非 ASCII コードの文字を使えるように考慮されています。特に、いくつかのメール転送プログラムでは、次のような現象が起きることがあります。

1. ヘッダ中の特定のフィールドのみを破棄する。
2. To:フィールドや Cc:フィールドのアドレスの順番を入れ換える。
3. フィールドの出現順序を入れ換える。
4. メッセージのヘッダが折り曲げられて、本来の位置とは違う場所に表示される。

さらに、いくつかのメールリーダでは、『(』、『[』、『{』などの文字に対しての RFC822 に従ったバックスラッシュを使ったクォーティングを、理解できずにパースができなかったり、使用頻度が低い機能を指定した場合にもパースできなかったりします。

不幸なことに、RFC822 のヘッダを正しく解釈できないこのようなプログラムを破棄するということは、インターネットでのメールシステムの運用に深刻な問題を引き起こすことになります。このメモで説明する拡張では、このような使用頻度が低い RFC822 の特殊な機能にも依存していません。

そのかわり、特定の 『ごく普通に』表示可能な (『encoded-word』としてひろく知られている) ASCII コードの文字列がエンコードされたデータの表現用に予約されています。encoded-word の文法は、通常のメッセージヘッダのテキスト中に 『偶然には』出現しにくいものにしてあります。しかも、encoded-word に使われる文字は、encoded-word が出現する場所で特別な意味を持たないように制限されます。

一般に 『encoded-word』は、表示可能な ASCII コードの文字からなる文字列で、『=?』で始まり、『?=』で終ります。また、その間に 2 つの 『?』があります。この文字列には、もとのテキストで使われている文字セットとエンコード方式、そしてテキストを印字可能な ASCII 文字列にエンコードしたものを記述します。

メールの編集プログラムは、非 ASCII の文字セットを使ったヘッダフィールドの入力を、encoded-word に変換してメッセージヘッダ (フィールドの適切な部分) に挿入します。

メールのリードプログラムは、このエンコードされた文字列が、メールヘッダの適切な部分に出現した場合にその記述を認識して、encoded-word 『そのもの』ではなくて、もとに戻したテキストを表示します。

注意

このメモは、STD 11、RFC 822、RFC 1521 に表記法や用語において強く依存しています。特に、このメモで使われる ABNF の構文は、STD 11、RFC 822 で定義されています。ヘッダの拡張に対しての文法で使われている用語についても同様です。このプロトコル拡張の正しい実装をするにあたって、STD 11、RFC 822、RFC 1521 の詳細について理解していることが要求されます。また、このメモで 『ASCII』と表記されるものは、『7-bit Ameriacan Standard Code for Information Interchange』, ANSI X3.4-1986 を暗黙的に指すものとします。MIME の charset 名でこの文字セットに対応するのは、『US-ASCII』です。この文書では MIME の charset 名を指す場合を除いて、簡潔さと STD 11、RFC 822 との一貫性のためにこの文字セットを表す用語として 『ASCII』を使います。実装者は、MIME のメッセージやボディ・パートのヘッダで指定する文字セット名には、必ず 『US-ASCII』と表記することに注意してください。

2 encode-word の構文

『encoded-word』は、RFC822 が表記法として使っている ABNF 文法で、次のように定義されています。この表記法は RFC 822 とほぼ同じですが、encoded-word のコンポーネントの間に空白文字が出現しない点で異なります。

```
encoded-word = "=" "?" charset "?" encoding "?" encoded-text "?" "="

charset = token ;  第 3 章を参照のこと。

encoding = token ; 第 4 章を参照のこと。

token = 1* <スペース、制御文字、especials を除いた任意の文字>
```

```
especials = "(" / ")" / "<" / ">" / "@" / "," / ";" / ":" / "\" /
<"> / "/" / "[" / "]" / "?" / "." / "="

encoded-text = 1*<表示可能な ASCII 文字で"?"以外のいずれか>
```
;(第 5 章 『メッセージヘッダでの encoded-word の使い方』
;を参照のこと)

「encoding」、「charset」に使う名前は、大小文字を問いません。つまり、文字セット名 「ISO-8859-1」と 「iso-8859-1」は同じものを指し、エンコーディング名 「Q」と 「q」は同じものを指すわけです。

encoded-word の長さは、文字セット、エンコード方式、エンコードされた文字列、デリミタを含めて、75 文字以内にします。テキストをエンコードした文字列の長さが 75 文字で収まりきらない場合には、(CRLF または空白文字で区切られた) 複数の encoded-word に分割します。ヘッダフィールドは、複数行にわたって記述できるので、複数の encoded-word を含むメッセージヘッダ行の 1 行の長さは、76 文字以内に制限されています。長さの制限は、ネットワーク間を結ぶメールゲートウェイでの処理を簡単にするだけでなく、ヘッダのパーサがトークンの解析をして encoded-word を見分ける場合に、(encoded-word の終了デリミタの?=を探すために) スキャンしなくてはならない文字数を限定できます。

エンコードされた文字列に出現する文字には、第 5 章で述べられている規則で制限が加えられています。

3 文字セット

encoded-word の 「charset」の部分には、エンコード前の文字列の文字セットを指定しています。この charset には、RFC 1521 での 「text/plain」ボディ・パートのパラメータ 「charset」に使える文字セット名、もしくは IANA で MIME の text/plain コンテントタイプ用に登録されている文字セットを使うことができます [3] (RFC 1521 の 7.1.1 節にある charsets を定義したリストを参照してください)。いくつかの文字セットでは、「ASCII モード」とそれ以外のモードとの間でモードを切り替えるコード切り替えのテクニックを使っています。 このような文字セットの文字列をエンコードした場合に、ASCII モードから他のモードに移行している場合には、encoded-word の最後に ASCII モードに復帰する制御コードを含むようにしてください。(この規則は、各 encoded-word 毎に適用され、ひとつのヘッダが複数に分けられて場合にもあてはまります)。

複数の文字セットの文字列が encoded-word にエンコードされる可能性がある場合で、かつ送信者と受信者との間に個人的なとりきめがなされていない場合には、ISO-8859-*シリーズのより上位の文字セットを選択するようにします。

4 エンコーディング

今のところ 「エンコード方式」として使えるのは、「Q」、「B」のみです。これらのエンコード方式については、後で説明します。「Q」エンコード方式を適用できるのは、エンコード後の文字列のほとんどが ASCII 文字セットにエンコードできる場合のみです。それ以外の場合には、「B」エンコード方式を適用すべきです。しかしながら、encoded-word の対応しているメールリーダでは、任意の文字セットの encoded-word に対して、両方のエンコード方式をサポートしている必要があります。

encoded-word として使えるのは、表示可能な ASCII 文字のサブセットのみです。encoded-word の区切りを明らかにするために、空白文字とタブ文字は encoded-word 内では使えません。「?」文字は、encoded-

word 内のパートの区切りとして使われます。したがって、エンコードテキスト (encoded-text) 内では使えません。その他にも、いくつかの文字が encoded-word 内で使うことができなくなっています。たとえば、From ヘッダでアドレスの記述の前にある「phrase」の部分には、RFC822 で規定されている「specials」に属する文字は、使うことができません。さらに、インターネットでメールゲートウェイ経由のメッセージの安全性のために使うことのできない文字もあります。

「B」エンコード方式では、これらの要求を自動的にみたしています。「Q」エンコード方式では、メッセージヘッダのなかでも影響を受けにくい場所 (たとえば、Subject フィールド) では、様々な表示可能な文字が使えますが、それ以外の場所では、ごく限られた文字のみが使えます。

4.1 B エンコード方式

B エンコード方式では、RFC1521 で規定されている「BASE64」エンコード方式と同じです。

4.2 Q エンコード方式

Q エンコード方式は、RFC1521 で content-transfer-encoding として規定されている「Quoted-Printable」とほぼ同じです。このエンコード方式は、ASCII 文字が多く含み、ASCII 端末上でデコードなしで解読できるようなテキストを生成するために考案されました。

1. 8 ビット値は、「=」で始まる 2 桁の 16 進数で表現されます。たとえば、"ISO-8859-1" 文字セットにおいて、「=」は「=3D」、空白文字は「=20」としてエンコードされます。(16 進数には、アルファベットの大文字を使います。)

2. 8 ビットの 16 進数で 20 に相当する文字 ("ISO-8859-1" では空白文字) は、「_」(ASCII 95 でのアンダースコア) で表現されます。(この文字を通さないネットワーク間メールゲートウェイも存在しますが、この文字を使うことで、このエンコード方式をサポートしていないメールリーダにおいて、「Q」エンコード方式を適用したデータの可読性を高めることができます。)「_」文字は、16 進コード 20 に相当する文字を表現したもので、スペースを違うコードに割り当てた文字セットでは、スペースを表現したものでないことに注意してください。

3. その文字セットに含まれる 8 ビット値で表現される文字のうち、「=」、「?」、「_」(アンダースコア)、空白以外で、表示可能な ASCII コード文字に対応するものがある場合には、その対応する文字を encoded-word として使います。(ただし、第 5 章にある制限について参照のこと)。

5 メッセージヘッダでの encoded-word の使い方

メッセージヘッダや、ボディヘッダで使われる encoded-word は、次のようなルールに従います。

1. encoded-word は、Subject または、Comments ヘッダフィールド、拡張ヘッダフィールド、フィールドボディが「*text」と定義されている RFC1521 のボディパート・フィールドの任意の場所で「text」トークンになります。encoded-word は、ユーザ定義の (「X-」) メッセージ、ボディパードのヘッダフィールドでも使えます。

   本来の ASCII からなる text と encoded-word は、同一のヘッダフィールドに共存できます。しかしながら、ヘッダフィールドに「*text」として出現する encoded-word は、隣接する encoded-word や「text」と連続した空白文字列で分割しなくてはいけません。

2. encoded-word は、「(」、「)」で仕切られたコメント、つまり「ctext」中の任意の場所でも使えます。もっと正確に RFC822 の ABNF 文法で「comment」を定義すると次のようになります。

```
comment = "(" * (ctext / quoted-pair / comment / encoded-word) ")"
```

comment 中に出現する「Q」エンコードされた encoded-word には、「(」、「)」、「\」を使えません。「comment」中の encoded-word は、隣接する encoded-word、「ctext」とは連続した空白文字列で仕切られていなくてはなりません。

3. From、To、Cc のフィールドでアドレスの前に記述される RFC822 での「word」のかわりに、「phrase」が定義されています。phrase は、ABNF で次のように定義されます。

```
phrase = 1*(encoded-word / word)
```

この使い方をする場合、「Q」エンコード方式の encoded-word は、次のような制限を受けます。

<ASCII コードの英大小文字、数字、「!」、「*」、「+」、「-」、「/」、「=」、「_」(ASCII 95 のアンダースコア)>

以上の場所でのみ encoded-word を使うことができます。特に「addr-spec」の中では、使えません。また、Received ヘッダフィールドでも使えません。

各 encoded-word は、8 ビット毎で 1 つの数にエンコードされなければなりません。各 encoded-word 中の encoded-text は、エンコード規則において正規のものでなければなりません。また次の encoded-word に続いていてはいけません。(たとえば「=?charset?Q?=?= =?charset?Q?AB?=」は、16 進数を表す「AB」が「=」の直後にないので、この規則に違反しています。)

各 encoded-word は、1 文字単位でエンコードされなければなりません。マルチバイト文字を使う場合には、1 つの文字が隣接する encoded-word にわたって分割されてはいけません。

これらのエンコード方式は、表示可能な文字と空白文字からなるテキストデータに対してのみ適用されるべきです。しかし、これらのエンコード方式は、8 ビットのデータをエンコードすることもできるので、デコードができるメールリーダを実装する場合には、受信者の端末で表示がされるときに不本意な副作用が起きないように保証すべきです。

非テキストのデータ (たとえば、絵や音声) をエンコードする場合の、この方式の適用についてはこのメモで定義されていません。純粋な ASCII 文字だけを表現する encoded-word も作ることができますが、普通はしません。encoded-word の形式にする必要があまりないからです。

6 メールリーダにおける encoded-word のサポート

6.1 メッセージヘッダ内での encoded-word の認識

メールリーダは RFC 822 の規則に従ってメッセージとボディパート・ヘッダをパースし、encoded-word を正しく認識しなくてはいけません。

encode-word は、以下のようにして認識されます。

1. 「text」として定義されるメッセージ、ボディパート・ヘッダ、ユーザ定義のヘッダフィールドは、以下のようにパースされます。まず、フィールドのボディの起点直後にある連続する空白文字列を除いた最長 75 文字の (空白文字を 1 つも含まない) 表示可能な文字列に対して、第 2 節で示した構文にあてはまる encoded-word があるかどうか調べます。この構文にあてはまらない表示可能な文字列については、字面どおりの ASCII 文字列とみなされます。

2. 「text」として定義されていないヘッダフィールドは、そのヘッダフィールド特有の構文規則でパースされます。しかし、「phrase」内に出現する「word」が第 2 節にある構文規則に合致する場合には、これを encoded-word として扱うべきです。それ以外の場合には、本来の「word」として扱います。

3. 「comment」中の、(空白を含まない) 最長 75 文字の表示可能な文字列で第 2 節の構文に合致するものがあれば、encoded-word として扱います。それ以外の場合は、通常の comment 文字列として扱います。

6.2 encoded-word の表示

デコードできる任意の encoded-word は、可能なかぎり元の文字セットで表示されます。

複数の encoded-word を含むようなフィールドの表示において、encoded-word の間にある連続する空白文字列は無視されます。(これにより、長いテキストをエンコードする際に、複数の encode-word 間に不必要な空白を狭むことなしに、これを表現することができます。)

将来、新たなエンコード方式が定義され、メールリーダがそのエンコーディングをサポートしていない場合には、次のどちらかをします。(a)encoded-word のテキストをエンコードされたまま表示する。(b) デコードできないことを示すメッセージを表示する。

メールリーダのサポートしていない文字セットが使われている場合には、次のどちらかが行なわれます。(a)(ヘッダにある)encoded-word のテキストをエンコードされたまま表示する。(b) できうる限りの「努力」をして表示できる分だけでも表示する。(c) デコードした文字列が表示できないことを示すメッセージを表示する。

コード切り替えのテクニックを使っている文字セットでは、エンコードされている文字列の表示は、暗黙的に「ASCII モード」から始めます。それに加えてメールリーダは、各 encoded-word の表示を終えたら、「ASCII モード」に戻っていることを保証しなくてはなりません。

6.3 正しい形式でない encoded-word のメールリーダでの取り扱い

第 2 節で定義されている構文に従う encoded-word には、適用されたエンコード方式の規則により、形式のうえで正しくないものがありえます。

1. 特定のエンコード方式において有効でない文字 (たとえば、「B」エンコーディングでの'-') を含んでいて正しくないもの。

2. encoded-word において、文字もしくは、8 ビットのデータ毎にエンコードがなされていないもの。

メールリーダは、正しい形式になっていない encoded-word を無理に表示する必要はありません。しかし、たとえそれが正しい形式になっていなくとも、メールリーダはそのメッセージ全体の表示や処理をやめてはいけません。

7　規則

この仕様書の要求をみたすメール編集プログラムは、『*text』、『*ctext』内の文字で『=?』で始まり、『=?』で終る空白文字列でない表示可能な ASCII 文字列が、有効な encoded-word であることを保証する必要があります。(ここで言う『始まる』とは、フィールドボディの起点と、空白文字列の直後の状態をさし、『終る』とはフィールドボディの終りと、空白文字列の直前の状態をさします。)

この仕様書の要求をみたすメールを読むためのプログラムは、メッセージヘッダの適切な場所に出現した encoded-word を第 6 節で定義した規則により『text』、『ctext』、『word』を区別できる必要があります。サポートする文字セットについては、『B』と『Q』両方のエンコーディング方式をサポートしていなければなりません。またプログラムは、デコードしたテキストの文字セットが『US-ASCII』である場合には、表示できなくてはなりません。メールを読むためのプログラムは、"ISO-8859-*" の文字セットについてもその中で ASCII 文字に該当する文字のある文字については、すくなくとも表示できるようになっていなくてはなりません。

8　例

```
From: =?US-ASCII?Q?Keith_Moore?= <moore@cs.utk.edu>
To: =?ISO-8859-1?Q?Keld_J=F8rn_Simonsen?= <keld@dkuug.dk>
CC: =?ISO-8859-1?Q?Andr=E9_?= Pirard <PIRARD@vm1.ulg.ac.be>
Subject: =?ISO-8859-1?B?SWYgeW91IGNhbiByZWFkIHRoaXMgeW8=?=
=?ISO-8859-2?B?dSB1bmRlcnN0YW5kIHRoZSBleGFtcGxlLg==?=
From: =?ISO-8859-1?Q?Olle_J=E4rnefors?= <ojarnef@admin.kth.se>
To: ietf-822@dimacs.rutgers.edu, ojarnef@admin.kth.se
Subject: Time for ISO 10646?

To: Dave Crocker <dcrocker@mordor.stanford.edu>
Cc: ietf-822@dimacs.rutgers.edu, paf@comsol.se
From: =?ISO-8859-1?Q?Patrik_F=E4ltstr=F6m?= <paf@nada.kth.se>
Subject: Re: RFC-HDR care and feeding

From: Nathaniel Borenstein <nsb@thumper.bellcore.com>
(=?iso-8859-8?b?7eXs+SDv4SDp7Oj08A==?=)
To: Greg Vaudreuil <gvaudre@NRI.Reston.VA.US>, Ned Freed
<ned@innosoft.com>, Keith Moore <moore@cs.utk.edu>
Subject: Test of new header generator
MIME-Version: 1.0
Content-type: text/plain; charset=ISO-8859-1
```

参考文献

[1] Borenstein N., and N. Freed, “MIME(Multipurpose Internet Mail Extensions) Part One: Mechanisms for Specifying and Describing the Format of Internet Message Bodies”, RFC 1521, Bellcore, Innosoft, September 1993.

[2] Crocker, D., “Standard for the Format of ARPA Internet Text Messages”, STD 11, RFC 822, UDEL, August 1982.

[3] Reynolds, J., and J. Postel, ”Assigned Numbers”, STD 2, RFC 1340, USC/Information Sciences Institute, July 1992.

9　セキュリティに関する考察

このメモでは、議論されていません。

10　著者連絡先

Keith Moore
University of Tennessee
107 Ayres Hall
Knoxville TN 37996-1301
Email: moore@cs.utk.edu