

Network Working Group
Request for Comments: 2557
Obsoletes: 2110
Category: Standards Track

J. Palme
Stockholm University/KTH
A. Hopmann
Microsoft Corporation
N. Shelness
Lotus Development Corporation
March 1999

MIME Encapsulation of Aggregate Documents, such as HTML (MHTML)

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.

Abstract

HTML [RFC 1866] defines a powerful means of specifying multimedia documents. These multimedia documents consist of a text/html root resource (object) and other subsidiary resources (image, video clip, applet, etc. objects) referenced by Uniform Resource Identifiers (URIs) within the text/html root resource. When an HTML multimedia document is retrieved by a browser, each of these component resources is individually retrieved in real time from a location, and using a protocol, specified by each URI.

In order to transfer a complete HTML multimedia document in a single e-mail message, it is necessary to: a) aggregate a text/html root resource and all of the subsidiary resources it references into a single composite message structure, and b) define a means by which URIs in the text/html root can reference subsidiary resources within that composite message structure.

This document a) defines the use of a MIME multipart/related structure to aggregate a text/html root resource and the subsidiary resources it references, and b) specifies a MIME content-header (Content-Location) that allow URIs in a multipart/related text/html root body part to reference subsidiary resources in other body parts of the same multipart/related structure.

While initially designed to support e-mail transfer of complete multi-resource HTML multimedia documents, these conventions can also be employed to resources retrieved by other transfer protocols such as HTTP and FTP to retrieve a complete multi-resource HTML multimedia document in a single transfer or for storage and archiving of complete HTML-documents.

Differences between this and a previous version of this standard, which was published as RFC 2110, are summarized in chapter 12.

Table of Contents

1. Introduction	3
2. Terminology	4
2.1 Conformance requirement terminology	4
2.2 Other terminology	4
3. Overview	6
4. The Content-Location MIME Content Header	6
4.1 MIME content headers	6
4.2 The Content-Location Header	7
4.3 URIs of MHTML aggregates	8
4.4 Encoding and decoding of URIs in MIME header fields	8
5. Base URIs for resolution of relative URIs	9
6. Sending documents without linked objects	10
7. Use of the Content-Type "multipart/related"	11
8. Usage of Links to Other Body Parts	13
8.1 General principle	13
8.2 Resolution of URIs in text/html body parts	13
8.3 Use of the Content-ID header and CID URLs	14
9. Examples	14
9.1 Example of a HTML body without included linked objects ...	15
9.2 Example with an absolute URI to an embedded GIF picture ..	15
9.3 Example with relative URIs to embedded GIF pictures	16
9.4 Example with a relative URI and no BASE available	17
9.5 Example using CID URL and Content-ID header to an embedded GIF picture	18
9.6 Example showing permitted and forbidden references between nested body parts	19
10. Character encoding issues and end-of-line issues	21
11. Security Considerations	22
11.1 Security considerations not related to caching	22
11.2 Security considerations related to caching	23
12. Differences as compared to the previous version of this proposed standard in RFC 2110	24
13. Acknowledgments	24
14. References	25
15. Authors' Addresses	27
16. Full Copyright Statement	28

1. Introduction

There are a number of document formats (Hypertext Markup Language [HTML2], Extended Markup Language [XML], Portable Document format [PDF] and Virtual Reality Markup Language [VRML]) that specify documents consisting of a root resource and a number of distinct subsidiary resources referenced by URIs within that root resource. There is an obvious need to be able to send such multi-resource documents in e-mail [SMTP], [RFC822] messages.

The standard defined in this document specifies how to aggregate such multi-resource documents in MIME-formatted [MIME1 to MIME5] messages for precisely this purpose.

While this specification was developed to satisfy the specific aggregation requirements of multi-resource HTML documents, it may also be applicable to other multi-resource document representations linked by URIs. While this is the case, there is no requirement that implementations claiming conformance to this standard be able to handle any URI linked document representations other than those whose root is HTML.

This aggregation into a single message of a root resource and the subsidiary resources it references may also be applicable to resources retrieved by other protocols such as HTTP or FTP, or to the archiving of complete web pages as they appeared at a particular point in time.

An informational RFC will be published as a supplement to this standard. The informational RFC will discuss implementation methods and some implementation problems. Implementers are strongly recommended to read this informational RFC when developing implementations of this standard. You can find it through URL <http://www.dsv.su.se/~jpalme/ietf/mhtml.html>.

This standard specifies that body parts to be referenced can be identified either by a Content-ID (containing a Message-ID value) or by a Content-Location (containing an arbitrary URL). The reason why this standard does not only recommend the use of Content-ID-s is that it should be possible to forward existing web pages via e-mail without having to rewrite the source text of the web pages. Such rewriting has several disadvantages, one of them that security checksums will probably be invalidated.

2. Terminology

2.1 Conformance requirement terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [IETF-TERMS].

An implementation is not compliant if it fails to satisfy one or more of the MUST requirements for the protocols it implements. An implementation that satisfies all the MUST and all the SHOULD requirements for its protocols is said to be "unconditionally compliant"; one that satisfies all the MUST requirements but not all the SHOULD requirements for its protocols is said to be "conditionally compliant."

2.2 Other terminology

Most of the terms used in this document are defined in other RFCs.

Absolute URI, AbsoluteURI	See Relative Uniform Resource Locators [RELURL].
CID	See Message/External Body Content-ID [MIDCID].
Content-Base	This header was specified in RFC 2110, but has been removed in this new version of the MHTML standard.
Content-ID	See Message/External Body Content-ID [MIDCID].
Content-Location	MIME message or content part header with one URI of the MIME message or content part body, defined in section 4.2 below.
Content-Transfer-Encoding	Conversion of a text into 7-bit octets as specified in [MIME1] chapter 6.
CR	See [RFC822].
CRLF	See [RFC822].
Displayed text	The text shown to the user reading a document with a web browser. This may be different from the HTML markup, see the definition of HTML markup below.

Header	Field in a message or content heading specifying the value of one attribute.
Heading	Part of a message or content before the first CRLF, containing formatted fields with attributes of the message or content.
HTML	See HTML 2 specification [HTML2].
HTML Aggregate objects	HTML objects together with some or all objects, to which the HTML object contains hyperlinks, directly or indirectly.
HTML markup	A file containing HTML encodings as specified in [HTML] which may be different from the displayed text which a person using a web browser sees. For example, the HTML markup may contain "<" where the displayed text contains the character "<".
LF	See [RFC822].
MIC	Message Integrity Codes, codes use to verify that a message has not been modified.
MIME	See the MIME specifications [MIME1 to MIME5].
MUA	Messaging User Agent.
PDF	Portable Document Format, see [PDF].
Relative URI, RelativeURI	See HTML 2 [HTML2] and RFC 1808 [RELURL].
URI, absolute and relative	See RFC 1866 [HTML2].
URL	See RFC 1738 [URL].
URL, relative	See Relative Uniform Resource Locators [RELURL].
VRML	See Virtual Reality Markup Language [VRML].

3. Overview

An aggregate document is a MIME-encoded message that contains a root resource (object) as well as other resources linked to it via URIs. These other resources may be required to display a multimedia document based on the root resource (inline pictures, style sheets, applets, etc.), or be the root resources of other multimedia documents. It is important to keep in mind that aggregate documents need to satisfy the differing needs of several audiences.

Mail sending agents might send aggregate documents as an encoding of normal day-to-day electronic mail. Mail sending agents might also send aggregate documents when a user wishes to mail a particular document from the web to someone else. Finally mail sending agents might send aggregate documents as automatic responders, providing access to WWW resources for non-IP connected clients. Also with other protocols such as HTTP or FTP, there may sometimes be a need to retrieve aggregate documents. Receiving agents also have several differing needs. Some receiving agents might be able to receive an aggregate document and display it just as any other text content type would be displayed. Others might have to pass this aggregate document to a browsing program, and provisions need to be made to make this possible.

Finally several other constraints on the problem arise. It is important that it be possible for a document to be signed and for it to be transmitted and displayed without breaking the message integrity (MIC) checksum that is part of the signature.

4. The Content-Location MIME Content Header

4.1 MIME content headers

In order to resolve URI references to resources in other body parts, one MIME content header is defined, Content-Location. This header can occur in any message or content heading.

The syntax for this header is, using the syntax definition tools from [ABNF]:

```
quoted-pair    =  ("\\" text)
text           =  %d1-9 / ; Characters excluding CR and LF
                %d11-12 /
                %d14-127
WSP           =  SP / HTAB ; Whitespace characters
```

```

FWS          =  ([*WSP CRLF] 1*WSP) ; Folding white-space

ctext        =  NO-WS-CTL / ; Non-white-space controls
               %d33-39 / ; The rest of the US-ASCII
               %d42-91 / ; characters not including "(",
               %d93-127 ; ")", or "\"

comment      =  "(" *([FWS] (ctext / quoted-pair / comment))
               [FWS] ")"

CFWS         =  *([FWS] comment) (([FWS] comment) / FWS)

content-location =  "Content-Location:" [CFWS] URI [CFWS]

URI          =  absoluteURI | relativeURI

```

where URI is restricted to the syntax for URLs as defined in Uniform Resource Locators [URL] until IETF specifies other kinds of URIs.

4.2 The Content-Location Header

A Content-Location header specifies an URI that labels the content of a body part in whose heading it is placed. Its value CAN be an absolute or a relative URI. Any URI or URL scheme may be used, but use of non-standardized URI or URL schemes might entail some risk that recipients cannot handle them correctly.

An URI in a Content-Location header need not refer to an resource which is globally available for retrieval using this URI (after resolution of relative URIs). However, URI-s in Content-Location headers (if absolute, or resolvable to absolute URIs) SHOULD still be globally unique.

A Content-Location header can thus be used to label a resource which is not retrievable by some or all recipients of a message. For example a Content-Location header may label an object which is only retrievable using this URI in a restricted domain, such as within a company-internal web space. A Content-Location header can even contain a fictitious URI. Such an URI need not be globally unique.

A single Content-Location header field is allowed in any message or content heading, in addition to a Content-ID header (as specified in [MIME1]) and, in Message headings, a Message-ID (as specified in [RFC822]). All of these constitute different, equally valid body part labels, and any of them may be used to satisfy a reference to a body part. Multiple Content-Location header fields in the same message heading are not allowed.

Example of a multipart/related structure containing body parts with both Content-Location and Content-ID labels:

```
Content-Type: multipart/related; boundary="boundary-example";
             type="text/html"

--boundary-example

Content-Type: text/html; charset="US-ASCII"

... .. <IMG SRC="fiction1/fiction2"> ... ..
... .. <IMG SRC="cid:97116092811xyz@foo.bar.net"> ... ..

--boundary-example
Content-Type: image/gif
Content-ID: <97116092511xyz@foo.bar.net>
Content-Location: fiction1/fiction2

--boundary-example
Content-Type: image/gif
Content-ID: <97116092811xyz@foo.bar.net>
Content-Location: fiction1/fiction3

--boundary-example--
```

4.3 URIs of MHTML aggregates

The URI of an MHTML aggregate is not the same as the URI of its root. The URI of its root will directly retrieve only the root resource itself, even if it may cause a web browser to separately retrieve in-line linked resources. If a Content-Location header field is used in the heading of a multipart/related, this Content-Location SHOULD apply to the whole aggregate, not to its root part.

When an URI referring to an MHTML aggregate is used to retrieve this aggregate, the set of resources retrieved can be different from the set of resources retrieved using the Content-Locations of its parts. For example, retrieving an MHTML aggregate may return an old version, while retrieving the root URI and its in-line linked objects may return a newer version.

4.4 Encoding and decoding of URIs in MIME header fields

4.4.1 Encoding of URIs containing inappropriate characters

Some documents may contain URIs with characters that are inappropriate for an RFC 822 header, either because the URI itself has an incorrect syntax according to [URL] or the URI syntax standard

has been changed to allow characters not previously allowed in MIME headers. These URIs cannot be sent directly in a message header. If such a URI occurs, all spaces and other illegal characters in it must be encoded using one of the methods described in [MIME3] section 4. This encoding MUST only be done in the header, not in the HTML text. Receiving clients MUST decode the [MIME3] encoding in the heading before comparing URIs in body text to URIs in Content-Location headers.

The charset parameter value "US-ASCII" SHOULD be used if the URI contains no octets outside of the 7-bit range. If such octets are present, the correct charset parameter value (derived e.g. from information about the HTML document the URI was found in) SHOULD be used. If this cannot be safely established, the value "UNKNOWN-8BIT" [RFC 1428] MUST be used.

Note, that for the matching of URIs in text/html body parts to URIs in Content-Location headers, the value of the charset parameter is irrelevant, but that it may be relevant for other purposes, and that incorrect labeling MUST, therefore, be avoided. Warning: Irrelevance of the charset parameter may not be true in the future, if different character encodings of the same non-English filename are used in HTML.

4.4.2 Folding of long URIs

Since MIME header fields have a limited length and long URIs can result in Content-Location headers that exceed this length, Content-Location headers may have to be folded.

Encoding as discussed in clause 4.4.1 MUST be done before such folding. After that, the folding can be done, using the algorithm defined in [URLBODY] section 3.1.

4.4.3 Unfolding and decoding of received URLs in MIME header fields

Upon receipt, folded MIME header fields should be unfolded, and then any MIME encoding should be removed, to retrieve the original URI.

5. Base URIs for resolution of relative URIs

Relative URIs inside the contents of MIME body parts are resolved relative to a base URI using the methods for resolving relative URIs described in [RELURL]. In order to determine this base URI, the first-applicable method in the following list applies.

- (a) There is a base specification inside the MIME body part containing the relative URI which resolves relative URIs into absolute URIs. For example, HTML provides the BASE element for this purpose.
- (b) There is a Content-Location header in the immediately surrounding heading of the body part and it contains an absolute URI. This URI can serve as a base in the same way as a requested URI can serve as a base for relative URIs within a file retrieved via HTTP [HTTP].
- (c) If necessary, step (b) can be repeated recursively to find a suitable Content-Location header in a surrounding multi-part or message heading.
- (d) If the MIME object is returned in a HTTP response, use the URI used to initiate the request
- (e) When the methods above do not yield an absolute URI, a base URL of "thismessage:/" MUST be employed. This base URL has been defined for the sole purpose of resolving relative references within a multipart/related structure when no other base URI is specified.

This is also described in other words in section 8.2 below.

6. Sending documents without linked objects

If a text/html resource (object) is sent without subsidiary resources, to which it refers, it MAY be sent by itself. In this case, embedding it in a multipart/related structure is not necessary.

Such a text/html resource may either contain no URIs, or URIs which the recipient is expected to retrieve (if possible) via a URI specified protocol. A text/html resource may also be sent with unresolvable links in special cases, such as when two authors exchange drafts of unfinished resources.

Inclusion of URIs referencing resources which the recipient has to retrieve via an URI specified protocol may not work for some recipients. This is because not all e-mail recipients have full Internet connectivity, or because URIs which work for a sender will not work for a recipient. This occurs, for example, when an URI refers to a resource within a company-internal network that is not accessible from outside the company.

7. Use of the Content-Type "multipart/related"

If a message contains one or more MIME body parts containing URIs and also contains as separate body parts, resources, to which these URIs (as defined, for example, in HTML 2.0 [HTML2]) refer, then this whole set of body parts (referring body parts and referred-to body parts) SHOULD be sent within a multipart/related structure as defined in [REL].

Even though headers can occur in a message that lacks an associated multipart/related structure, this standard only covers their use for resolution of URIs between body parts inside a multipart/related structure. This standard does cover the case where a resource in a nested multipart/related structure contains URIs that reference MIME body parts in another multipart/related structure, in which it is enclosed. This standard does not cover the case where a resource in a multipart/related structure contains URIs that reference MIME body parts in another parallel or nested multipart/related structure, or in another MIME message, even if methods similar to those described in this standard are used. Implementers who employ such URIs are warned that receiving agents implementing this standard may not be able to process such references.

When the start body part of a multipart/related structure is an atomic object, such as a text/html resource, it SHOULD be employed as the root resource of that multipart/related structure. When the start body part of a multipart/related structure is a multipart/alternative structure, and that structure contains at least one alternative body part which is a suitable atomic object, such as a text/html resource, then that body part SHOULD be employed as the root resource of the aggregate document. Implementers are warned, however, that some receiving agents treat multipart/alternative as if it had been multipart/mixed (even though MIME [MIME1] requires support for multipart/alternative).

[REL] specifies that a type parameter is mandatory in a "Content-Type: multipart/related" header, and requires that it be employed to specify the type of the multipart/related start object. Thus, the type parameter value shall be "multipart/alternative", when the start part is of "Content-type multipart/alternative", even if the actual root resource is of type "text/html". In addition, if the multipart/related start object is not the first body part in a multipart/related structure, [REL] further requires that its Content-ID MUST be specified as the value of a start parameter in the "Content-Type: multipart/related" header.

When rendering a resource in a multipart/related structure, URI references within that resource can be satisfied by body parts within the same multipart/related structure (see section 8.2 below). This is useful:

- (a) For those recipients who only have email but not full Internet access.
- (b) For those recipients who for other reasons, such as firewalls or the use of company-internal links, cannot retrieve URI referenced resources via URI specified protocols.

Note, that this means that you can, via e-mail, send text/html objects which includes URIs which the recipient cannot resolve via HTTP or other connectivity-requiring URIs.

- (c) To send a document whose content is preserved even if the resources to which embedded URIs refer are later changed or deleted.
- (d) For resources which are not available for protocol based retrieval.
- (e) To speed up access.

When a sending MUA sends objects which were retrieved from the WWW, it SHOULD maintain their WWW URIs. It SHOULD not transform these URIs into some other URI form prior to transmitting them. This will allow

the receiving MUA to both verify MICs included with the message, as well as verify the documents against their WWW counterpoints, if this is appropriate.

In certain cases this will not work - for example, if a resource contains URIs as parameters to objects and applets. In such a case, it might be better to rewrite the document before sending it. This problem is discussed in more detail in the informational RFC which will be published as a supplement to this standard.

Within a multipart/related structure, each body part MUST have, if assigned, a different Content-ID header value and a Content-Location header field values which resolve to a different URI.

Two body parts in the same multipart/related structure can have the same relative Content-Location header value, only if when resolved to absolute URIs they become different.

8. Usage of Links to Other Body Parts

8.1 General principle

A body part, such as a text/html body part, may contain URIs that reference resources which are included as body parts in the same message -- in detail, as body parts within the same multipart/related structure. Often such URI linked resources are meant to be displayed inline to the viewer of the referencing body part; for example, objects referenced with the SRC attribute of the IMG element in HTML 2.0 [HTML2]. New elements and attributes with this property are proposed in the ongoing development of HTML (examples: applet, frame, profile, OBJECT, classid, codebase, data, SCRIPT). A sender might also want to send a set of HTML documents which the reader can traverse, and which are related with the attribute href of the A element.

If a user retrieves and displays a web page formed from a text/html resource, and the subsidiary resources it references, and merely saves the text/html resource, that user may not at a later time be able to retrieve and display the web page as it appeared when saved. The format described in this standard can be used to archive and retrieve all of the resources required to display the web page, as it originally appeared at a certain moment of time, in one aggregate file.

In order to send or store complete such messages, there is a need to specify how a URI in one body part can reference a resource in another body part.

8.2 Resolution of URIs in text/html body parts

The resolution of inline, retrieval and other kinds of URIs in text/html body parts is performed in the following way:

- (a) Unfold multiple line header values according to [URLBODY]. Do NOT however translate character encodings of the kind described in [URL]. Example: Do not transform "a%2eb/c%20d" into "a/b/c d".
- (b) Remove all MIME encodings, such as content-transfer encoding and header encodings as defined in MIME part 3 [MIME3] Do NOT however translate character encodings of the kind described in [URL]. Example: Do not transform "a%2eb/c%20d" into "a/b/c d".
- (c) Try to resolve all relative URIs in the HTML content and in Content-Location headers using the procedure described in chapter 5 above. The result of this resolution can be an absolute URI, or an absolute URI with the base "thismessage:/" as specified in

chapter 5.

- (d) For each referencing URI in a text/html body part, compare the value of the referencing URI after resolution as described in (a) and (b), with the URI derived from Content-ID and Content-Location headers for other body parts within the same or a surrounding Multipart/related structure. If the strings are identical, octet by octet, then the referencing URI references that body part. This comparison will only succeed if the two URIs are identical. This means that if one of the two URIs to be compared was a fictitious absolute URI with the base "thismessage:/", the other must also be such a fictitious absolute URI, and not resolvable to a real absolute URI.
- (e) If (d) fails, try to retrieve the URI referenced resource hyperlink through ordinary Internet lookup. Resolution of URIs of the URL-types "mid" or "cid" to other content-parts, outside the same multipart/related structure, or in other separately sent messages, is not covered by this standard, and is thus neither encouraged nor forbidden.

8.3 Use of the Content-ID header and CID URLs

When URIs employing a CID (Content-ID) scheme as defined in [URL] and [MIDCID] are used to reference other body parts in an MHTML multipart/related structure, they MUST only be matched against Content-ID header values, and not against Content-Location header with CID: values. Thus, even though the following two headers are identical in meaning, only the Content-ID value will be matched, and the Content-Location value will be ignored.

```
Content-ID: <foo@bar.net>  
Content-Location: CID: foo@bar.net
```

Note: Content-IDs MUST be globally unique [MIME1]. It is thus not permitted to make them unique only within a message or within a single multipart/related structure.

9. Examples

Warning: The examples are provided for illustrative purposes only. If there is a contradiction between the explanatory text and the examples in this standard, then the explanatory text is normative.

Notation: The examples contain indentation to show the structure, the real objects should not be indented in this way.

9.1 Example of a HTML body without included linked objects

The first example is the simplest form of an HTML email message. This message does not contain an aggregate HTML object, but simply a message with a single HTML body part. This body part contains a URI but the messages does not contain the resource referenced by that URI. To retrieve the resource referenced by the URI the receiving client would need either IP access to the Internet, or an electronic mail web gateway.

```
From: fool@bar.net
To: foo2@bar.net
Subject: A simple example
Mime-Version: 1.0
Content-Type: text/html; charset="iso-8859-1"
Content-Transfer-Encoding: 8bit
```

```
<HTML>
<head></head>
<body>
<h1>Acute accent</h1>
The following two lines look have the same screen rendering:<p>
E with acute accent becomes É.<br>
E with acute accent becomes &Eacute;.<p>
Try clicking <a href="http://www.ietf.cnri.reston.va.us/">
here.</a><p>
</body></HTML>
```

9.2 Example with an absolute URI to an embedded GIF picture

The second example is an HTML message which includes a single image, referenced using the Content-Location mechanism.

```
From: fool@bar.net
To: foo2@bar.net
Subject: A simple example
Mime-Version: 1.0
Content-Type: multipart/related; boundary="boundary-example";
             type="text/html"; start="<foo3@fool@bar.net>"
```

```
--boundary-example
Content-Type: text/html; charset="US-ASCII"
Content-ID: <foo3@fool@bar.net>
```

```
... text of the HTML document, which might contain a URI
referencing a resource in another body part, for example
through a statement such as:
<IMG SRC="http://www.ietf.cnri.reston.va.us/images/ietflogo.gif"
```

```

    ALT="IETF logo">

--boundary-example
Content-Location:
    http://www.ietf.cnri.reston.va.us/images/ietflogo.gif
Content-Type: IMAGE/GIF
Content-Transfer-Encoding: BASE64

R0lGODlhGAGgAPEAAP/////ZRACgoAAAACH+PUNvcHlyaWdodCAoQykgMTk5
NSBJRVRGLiBVbmFldGhvcml6ZWQgZHVwbGljYXRpb24gcHJvaGliaXRlZC4A
etc...

--boundary-example--

```

9.3 Example with relative URIs to embedded GIF pictures

In this example, a Content-Location header field in the outermost heading will be a base to all relative URLs, also inside the HTML text being sent.

```

From: fool@bar.net
To: foo2@bar.net
Subject: A simple example
Mime-Version: 1.0
Content-Location: http://www.ietf.cnri.reston.va.us/
Content-Type: multipart/related; boundary="boundary-example";
    type="text/html"

```

```

--boundary-example
Content-Type: text/html; charset="ISO-8859-1"
Content-Transfer-Encoding: QUOTED-PRINTABLE

```

... text of the HTML document, which might contain URIs referencing resources in other body parts, for example through statements such as:

```

<IMG SRC="images/ietflogo1.gif" ALT="IETF logo1">
<IMG SRC="images/ietflogo2.gif" ALT="IETF logo2">
<IMG SRC="images/ietflogo3.gif" ALT="IETF logo3">

```

Example of a copyright sign encoded with Quoted-Printable: =A9
 Example of a copyright sign mapped onto HTML markup: ¨

```

--boundary-example
Content-Location:
    http://www.ietf.cnri.reston.va.us/images/ietflogo1.gif
; Note - Absolute Content-Location does not require a
; base

```

```
Content-Type: IMAGE/GIF
Content-Transfer-Encoding: BASE64
```

```
R0lGODlhGAGgAPEAAP/////ZRaCgoAAAACH+PUNvchlyaWdodCAoQykgMTk5
NSBJRVRGLiBVbmFldGhvcml6ZWQgZHVwbGljYXRpb24gcHJvaGliaXRlZC4A
etc...
```

```
--boundary-example
Content-Location: images/ietflogo2.gif
; Note - Relative Content-Location is resolved by base
; specified in the Multipart/Related Content-Location heading
Content-Transfer-Encoding: BASE64
```

```
R0lGODlhGAGgAPEAAP/////ZRaCgoAAAACH+PUNvchlyaWdodCAoQykgMTk5
NSBJRVRGLiBVbmFldGhvcml6ZWQgZHVwbGljYXRpb24gcHJvaGliaXRlZC4A
etc...
```

```
--boundary-example
Content-Location:
    http://www.ietf.cnri.reston.va.us/images/ietflogo3.gif
Content-Transfer-Encoding: BASE64
```

```
R0lGODlhGAGgAPEAAP/////ZRaCgoAAAACH+PUNvchlyaWdodCAoQykgMTk5
NSBJRVRGLiBVbmFldGhvcml6ZWQgZHVwbGljYXRpb24gcHJvaGliaXRlZC4A
etc...
```

```
--boundary-example--
```

9.4 Example with a relative URI and no BASE available

```
From: fool@bar.net
To: foo2@bar.net
Subject: A simple example
Mime-Version: 1.0
Content-Type: multipart/related; boundary="boundary-example";
    type="text/html"
```

```
--boundary-example
Content-Type: text/html; charset="iso-8859-1"
Content-Transfer-Encoding: QUOTED-PRINTABLE
```

```
... text of the HTML document, which might contain a URI
referencing a resource in another body part, for example
through a statement such as:
<IMG SRC="ietflogo.gif" ALT="IETF logo">
Example of a copyright sign encoded with Quoted-Printable: =A9
Example of a copyright sign mapped onto HTML markup: &#168;
```

```
--boundary-example
Content-Location: ietflogo.gif
Content-Type: IMAGE/GIF
Content-Transfer-Encoding: BASE64

R0lGODlhGAGgAPEAAP/////ZRACgoAAAACH+PUNvcHlyaWdodCAoQykgMTk5
NSBJRVRGLiBVbmF1dGhvcml6ZWQgZHVwbGljYXRpb24gcHJvaGliaXRlZC4A
etc...

--boundary-example--
```

9.5 Example using CID URL and Content-ID header to an embedded GIF picture

```
From: fool@bar.net
To: foo2@bar.net
Subject: A simple example
Mime-Version: 1.0
Content-Type: multipart/related; boundary="boundary-example";
             type="text/html"
```

```
--boundary-example
Content-Type: text/html; charset="US-ASCII"
```

... text of the HTML document, which might contain a URI referencing a resource in another body part, for example through a statement such as:

```
<IMG SRC="cid:foo4@fool@bar.net" ALT="IETF logo">
```

```
--boundary-example
Content-Location: CID:something@else ; this header is disregarded
Content-ID: <foo4@fool@bar.net>
Content-Type: IMAGE/GIF
Content-Transfer-Encoding: BASE64
```

```
R0lGODlhGAGgAPEAAP/////ZRACgoAAAACH+PUNvcHlyaWdodCAoQykgMTk5
NSBJRVRGLiBVbmF1dGhvcml6ZWQgZHVwbGljYXRpb24gcHJvaGliaXRlZC4A
etc...
```

```
--boundary-example--
```

9.6 Example showing permitted and forbidden references between nested body parts

This example shows in which cases references are allowed between multiple multipart/related body parts in a message.

```
From: fool@bar.net
To: foo2@bar.net
Subject: A simple example
Mime-Version: 1.0
Content-Type: multipart/related; boundary="boundary-example-1";
             type="text/html"
```

```
--boundary-example-1
Content-Type: text/html;charset="US-ASCII"
Content-ID: <foo3@fool@bar.net>
```

The image reference below will be resolved with the image in the next body part.

```
<IMG SRC="http://www.ietf.cnri.reston.va.us/images/ietflogo.gif"
ALT="IETF logo with white background">
```

The image reference below cannot be resolved within this MIME message, since it contains a reference from an outside body part to an inside body part, which is not supported by this standard.

```
<IMG SRC=images/ietflogo2e.gif"
ALT="IETF logo with transparent background">
```

The anchor reference immediately below will be resolved with the nested text/html body part below:

```
<A HREF="http://www.ietf.cnri.reston.va.us/more-info"
More info</A>
```

The anchor reference immediately below will be resolved with the nested text/html body part below:

```
<A HREF="http://www.ietf.cnri.reston.va.us/even-more-info"
Even more info</A>
```

```
--boundary-example-1
Content-Location:
    http://www.ietf.cnri.reston.va.us/images/ietflogo.gif
Content-Type: IMAGE/GIF
Content-Transfer-Encoding: BASE64
```

```
R0lGODlhGAGgAPEAAP/////ZRaCgoAAAACH+PUNvcHlyaWdodCAoQykgMTk5
NSBJRVRGLiBVbmFldGhvcml6ZWQgZHVwbGljYXRpb24gcHJvaGlhXlZC4A
etc...
```

```
--boundary-example-1
Content-Location:
    http://www.ietf.cnri.reston.va.us/more-info
Content-Type: multipart/related; boundary="boundary-example-2";
    type="text/html"
--boundary-example-2
Content-Type: text/html;charset="US-ASCII"
Content-ID: <foo4@fool@bar.net>
```

The image reference below will be resolved with the image in the surrounding multipart/related above.

```
<IMG SRC="images/ietflogo.gif"
ALT="IETF logo with white background">
```

The image reference below will be resolved with the image inside the current nested multipart/related below.

```
<IMG SRC=images/ietflogo2e.gif"
ALT="IETF logo with transparent background">
```

```
--boundary-example-2
Content-Location: http:images/ietflogo2.gif
Content-Type: IMAGE/GIF
Content-Transfer-Encoding: BASE64
```

```
R0lGODlhGAGgANX/ACkpkTExMTk5OUJCQkpKSlJSUlpaWmNjY2tra3Nzc3t7e4
SEhIyMjJSUlJycnKWlpa2trbWltcDAwM7Ozv/eQnNzjHNzlGtrjGNjhFpaelpa
etc...
```

```
--boundary-example-2--
--boundary-example-1
Content-Location:
    http://www.ietf.cnri.reston.va.us/even-more-info
Content-Type: multipart/related; boundary="boundary-example-3";
    type="text/html"
--boundary-example-3
Content-Type: text/html;charset="US-ASCII"
Content-ID: <4@foo@bar.net>
```

The image reference below will be resolved with the image inside the current nested multipart/related below.

```
<IMG SRC=images/ietflogo2d.gif"
ALT="IETF logo with shadows">
```

The image reference below cannot be resolved according to this standard since references between parallel multipart/related structures are not supported.

```
<IMG SRC=images/ietflogo2e.gif"
ALT="IETF logo with transparent background">
```

```

--boundary-example-3
Content-Location: http:images/ietflogo2d.gif
Content-Type: IMAGE/GIF
Content-Transfer-Encoding: BASE64

R0lGODlhGAGgANX/AMDawCkpKTExMTk5OUJCQkpKSlJSUlpaWmNjY2tra3Nz
c3t7e4SEhIyMjJSUlJycnKWlpa2trbWltb29vcbGxs7OztbWlt7e3ufn5+/v
etc...

--boundary-example-3--
--boundary-example-1--

```

10. Character encoding issues and end-of-line issues

For the encoding of characters in HTML documents and other text documents into a MIME-compatible octet stream, the following mechanisms are relevant:

- HTML [HTML2], [HTML-I18N] as an application of SGML [SGML] allows characters to be denoted by character entities as well as by numeric character references (e.g. "Latin small letter a with acute accent" may be represented by "´" or "á") in the HTML markup.
- HTML documents, in common with other documents of the MIME Content-Type "text", can be represented in MIME using one of several character encodings. The MIME Content-Type "charset" parameter value indicates the particular encoding used. For the exact meaning and use of the "charset" parameter, please see [MIME2] chapter 4.

Note that the "charset" parameter refers only to the MIME character encoding. For example, the string "´" can be sent in MIME with "charset=US-ASCII", while the raw character "Latin small letter a with acute accent" cannot.

The above mechanisms are well defined and documented, and therefore not further explained here. In sending a message, all the above mentioned mechanisms MAY be used, and any mixture of them MAY occur when sending the document in MIME format. Receiving user agents (together with any Web browser they may use to display the document) MUST be capable of handling any combinations of these mechanisms.

Also note that:

- Any documents including HTML documents that contain octet values outside the 7-bit range need a content-transfer-encoding applied before transmission over certain transport protocols [MIME1,

chapter 5].

- The MIME standard [MIME2] requires that e-mailed documents of "Content-Type: Text/ MUST be in canonical form before a Content-Transfer-Encoding is applied, i.e. that line breaks are encoded as CRLFs, not as bare CRs or bare LFs or something else. This is in contrast to [HTTP] where section 3.6.1 allows other representations of line breaks.

Note that this might cause problems with integrity checks based on checksums, which might not be preserved when moving a document from the HTTP to the MIME environment. If a document has to be converted in such a way that a checksum based message integrity check becomes invalid, then this integrity check header SHOULD be removed from the document.

Other sources of problems are Content-Encoding used in HTTP but not allowed in MIME, and character sets that are not able to represent line breaks as CRLF. A good overview of the differences between HTTP and MIME with regards to Content-Type: "text" can be found in [HTTP], appendix C.

Some transport mechanisms may specify a default "charset" parameter if none is supplied [HTTP, MIME1]. Because the default differs for different mechanisms, when HTML is transferred through e-mail, the charset parameter SHOULD be included, rather than relying on the default.

11. Security Considerations

11.1 Security considerations not related to caching

It is possible for a message sender to misrepresent the source of a multipart/related body part to a message recipient by labeling it with a Content-Location URI that references another resource. Therefore, message recipients should only interpret Content-Location URIs as labeling a body part for the resolution of references from body parts in the same multipart/related message structure, and not as the source of a resource, unless this can be verified by other means.

URIs, especially File URIs, if used without change in a message, may inadvertently reveal information that was not intended to be revealed outside a particular security context. Message senders should take care when constructing messages containing the new header fields, defined in this standard, that they are not revealing information outside of any security contexts to which they belong.

Some resource servers hide passwords and tickets (access tokens to information which should not be revealed to others) and other sensitive information in non-visible fields or URIs within a text/html resource. If such a text/html resource is forwarded in an email message, this sensitive information may be inadvertently revealed to others.

Since HTML documents can either directly contain executable content (i.e., JavaScript) or indirectly reference executable content (The "INSERT" specification, Java). It is exceedingly dangerous for a receiving User Agent to execute content received in a mail message without careful attention to restrictions on the capabilities of that executable content.

HTML-formatted messages can be used to investigate user behaviour, for example to break anonymity, in ways which invade the privacy of individuals. If you send a message with an inline link to an object which is not itself included in the message, the recipients mailer or browser may request that object through HTTP. The HTTP transaction will then reveal who is reading the message. Example: A person who wants to find out who is behind an anonymous user identity, or from which workstation a user is reading his mail, can do this by sending a message with an inline link and then observe from where this link is used to request the object.

11.2 Security considerations related to caching

There is a well-known problem with the caching of directly retrieved web resources. A resource retrieved from a cache may differ from that re-retrieved from its source. This problem, also manifests itself when a copy of a resource is delivered in a multipart/related structure.

When processing (rendering) a text/html body part in an MHTML multipart/related structure, all URIs in that text/html body part which reference subsidiary resources within the same multipart/related structure SHALL be satisfied by those resources and not by resources from any another local or remote source.

Therefore, if a sender wishes a recipient to always retrieve an URI referenced resource from its source, an URI labeled copy of that resource MUST NOT be included in the same multipart/related structure.

In addition, since the source of a resource received in a multipart/related structure can be misrepresented (see 11.1 above), if a resource received in multipart/related structure is stored in a cache, it MUST NOT be retrieved from that cache other than by a

reference contained in a body part of the same multipart/related structure. Failure to honor this directive will allow a multipart/related structure to be employed as a Trojan Horse. For example, to inject bogus resources (i.e. a misrepresentation of a competitor's Web site) into a recipient's generally accessible Web cache.

12. Differences as compared to the previous version of this proposed standard in RFC 2110

The specification has been changed to show that the formats described do not only apply to multipart MIME in email, but also to multipart MIME transferred through other protocols such as HTTP or FTP.

In order to agree with [RELURL], Content-Location headers in multipart Content-Headings can now be used as a base to resolve relative URIs in their component parts, but only if no base URI can be derived from the component part itself. Base URIs in Content-Location header fields in inner headings have precedence over base URIs in outer multipart headings.

The Content-Base header, which was present in RFC 2110, has been removed. A conservative implementor may choose to accept this header in input for compatibility with implementations of RFC 2110, but MUST never send any Content-Base header, since this header is not any more a part of this standard.

A section 4.4.1 has been added, specifying how to handle the case of sending a body part whose URI does not agree with the correct URI syntax.

The handling of relative and absolute URIs for matching between body parts have been merged into a single description, by specifying that relative URIs, which cannot be resolved otherwise, should be handled as if they had been given the URL "thismessage:/".

13. Acknowledgments

Harald T. Alvestrand, Richard Baker, Isaac Chan, Dave Crocker, Martin J. Duerst, Lewis Geer, Roy Fielding, Ned Freed, Al Gilman, Paul Hoffman, Andy Jacobs, Richard W. Jesmajian, Mark K. Joseph, Greg Herlihy, Valdis Kletnieks, Daniel LaLiberte, Ed Levinson, Jay Levitt, Albert Lunde, Larry Masinter, Keith Moore, Gavin Nicol, Martyn W. Peck, Pete Resnick, Jon Smirl, Einar Stefferud, Jamie Zawinski, Steve Zilles and several other people have helped us with preparing this document. We alone take responsibility for any errors which may still be in the document.

14. References

- [ABNF] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", RFC 2234, November 1997.
- [CONDISP] Troost, R. and S. Dorner, "Communicating Presentation Information in Internet Messages: The Content-Disposition Header", RFC 2183, August 1997.
- [HOSTS] Braden, R., Ed., "Requirements for Internet Hosts -- Application and Support", STD 3, RFC 1123, October 1989.
- [HTML-I18N] Yergeau, F., Nicol, G. Adams, G. and M. Duerst: "Internationalization of the Hypertext Markup Language", RFC 2070, January 1997.
- [HTML2] Berners-Lee, T. and D. Connolly: "Hypertext Markup Language - 2.0", RFC 1866, November 1995.
- [HTML3.2] Dave Raggett: HTML 3.2 Reference Specification, W3C Recommendation, January 1997, at URL <http://www.w3.org/TR/REC-html32.html>
- [HTTP] Berners-Lee, T., Fielding, R. and H. Frystyk, "Hypertext Transfer Protocol -- HTTP/1.0", RFC 1945, May 1996.
- [IETF-TERMS] Bradner, S., "Key words for use in RFCs to Indicate Requirements Levels", BCP 14, RFC 2119, March 1997.
- [INFO] J. Palme: Sending HTML in MIME, an informational supplement to the RFC: MIME Encapsulation of Aggregate Documents, such as HTML (MHTML), Work in Progress.
- [MD5] Rivest, R., "The MD5 Message-Digest Algorithm", RFC 1321, April 1992.
- [MIDCID] Levinson, E., "Content-ID and Message-ID Uniform Resource Locators", RFC 2387, August 1998.
- [MIME1] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", RFC 2045, December 1996.

- [MIME2] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types", RFC 2046, December 1996.
- [MIME3] Moore, K., "MIME (Multipurpose Internet Mail Extensions) Part Three: Message Header Extensions for Non-ASCII Text", RFC 2047, December 1996.
- [MIME4] Freed, N., Klensin, J. and J. Postel, "Multipurpose Internet Mail Extensions (MIME) Part Four: Registration Procedures", RFC 2048, January 1997.
- [MIME5] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part Five: Conformance Criteria and Examples", RFC 2049, November 1996.
- [NEWS] Horton, M. and R. Adams: "Standard for interchange of USENET messages", RFC 1036, December 1987.
- [PDF] Tim Bienz and Richar Cohn: "Portable Document Format Reference Manual", Addison-Wesley, Reading, MA, USA, 1993, ISBN 0-201-62628-4.
- [REL] Levinson, E., "The MIME Multipart/Related Content-Type", RFC 2389, August 1998.
- [RELURL] Fielding, R., "Relative Uniform Resource Locators", RFC 1808, June 1995.
- [RFC822] Crocker, D., "Standard for the format of ARPA Internet text messages." STD 11, RFC 822, August 1982.
- [SGML] ISO 8879. Information Processing -- Text and Office - Standard Generalized Markup Language (SGML), 1986.
<URL:http://www.iso.ch/cate/d16387.html>
- [SMTP] Postel, J., "Simple Mail Transfer Protocol", STD 10, RFC 821, August 1982.
- [URL] Berners-Lee, T., Masinter, L. and M. McCahill, "Uniform Resource Locators (URL)", RFC 1738, December 1994.
- [URLBODY] Freed, N. and K. Moore, "Definition of the URL MIME External-Body Access-Type", RFC 2017, October 1996.

- [VRML] Gavin Bell, Anthony Parisi, Mark Pesce: "Virtual Reality Modeling Language (VRML) Version 1.0 Language Specification." May 1995, <http://www.vrml.org/Specifications/>.
- [XML] Extensible Markup Language, published by the World Wide Web Consortium, URL <http://www.w3.org/XML/>

15. Authors' Addresses

For contacting the editors, preferably write to Jacob Palme.

Jacob Palme
Stockholm University and KTH
Electrum 230
S-164 40 Kista, Sweden

Phone: +46-8-16 16 67
Fax: +46-8-783 08 29
EMail: jpalme@dsv.su.se

Alex Hopmann
Microsoft Corporation
One Microsoft Way
Redmond WA 98052

Phone: +1-425-703-8238
EMail: alexhop@microsoft.com

Nick Shelness
Lotus Development Corporation
55 Cambridge Parkway
Cambridge MA 02142-1295

EMail: Shelness@lotus.com

Working group chairman:

Einar Stefferud
EMail: stef@nma.com

16. 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.

