NAME

Mail::Message::Construct::Reply − reply to a Mail::Message

SYNOPSIS

```
 my Mail::Message $reply = $message−>reply;
 my $quoted  = $message−>replyPrelude($head−>get('From'));
```

DESCRIPTION

Complex functionality on Mail::Message objects is implemented in different files which are autoloaded. This file implements the functionality related to creating message replies.

METHODS

Constructing a message

$obj−>**reply**(%options)

Start a reply to this message. Some of the header-lines of the original message will be taken. A message-id will be assigned. Some header lines will be updated to facilitate message-thread detection (see Mail::Box::Thread::Manager).

You may reply to a whole message or a message part. You may wish to overrule some of the default header settings for the reply immediately, or you may do that later with `set` on the header.

ADDRESSES may be specified as string, or a Mail::Address object, or as array of Mail::Address objects.

All `%options` which are not listed below AND start with a capital, will be added as additional headers to the reply message.

```
 −Option          −−Default
  Bcc               undef
  Cc                <'cc' in current>
  From              <'to' in current>
  Message−ID        <uniquely generated>
  Subject           replySubject()
  To                <sender in current>
  body              undef
  group_reply       <true>
  include           'INLINE'
  max_signature     10
  message_type      Mail::Message
  postlude          undef
  prelude           undef
  quote             '> '
  signature         undef
  strip_signature   qr/ˆ−−\s/
```

Bcc => ADDRESSES
  Receivers of blind carbon copies: their names will not be published to other message receivers.

Cc => ADDRESSES
  The carbon-copy receivers, by default a copy of the `Cc` field of the source message.

From => ADDRESSES
  Your identification, by default taken from the `To` field of the source message.

Message-ID => STRING
  Supply a STRING as specific message-id for the reply. By default, one is generated for you. If there are no angles around your id, they will be added.

Subject => STRING|CODE
  Force the subject line to the specific STRING, or the result of the subroutine specified by CODE. The subroutine will be called passing the subject of the original message as only argument. By default,

Mail::Message::replySubject() is used.

To => ADDRESSES
The destination of your message. By default taken from the `Reply-To` field in the source message. If that field is not present as well, the`From` line is scanned. If the y all fail, `undef` is returned by this method: no reply message produced.

body => BODY
Usually, the reply method can create a nice, sufficient message from the source message's body. In case you like more complicated reformatting, you may also create a body yourself first, and pass this on to this `reply` method. Some of the other options to this method will be ignored in this case.

group_reply => BOOLEAN
Will the people listed in the `Cc` headers (those who received the message where you reply to now) also receive this message as carbon copy?

include => 'NO'|'INLINE'|'ATTACH'
Must the message where this is a reply to be included in the message? If `NO` then not. With `INLINE` a reply body is composed. `ATTACH` will create a multi-part body, where the original message is added after the specified body. It is only possible to inline textual messages, therefore binary or multipart messages will always be enclosed as attachment.

max_signature => INTEGER
Passed to `stripSignature` on the body as parameter `max_lines`. Only effective for single-part messages.

message_type => CLASS
Create a message with the requested type. By default, it will be a Mail::Message. This is correct, because it will be coerced into the correct folder message type when it is added to that folder.

postlude => BODY|LINES
The line(s) which to be added after the quoted reply lines. Create a body for it first. This should not include the signature, which has its own option. The signature will be added after the postlude when the reply is INLINEd.

prelude => BODY|LINES
The line(s) which will be added before the quoted reply lines. If nothing is specified, the result of the **replyPrelude()** method is taken. When`undef` is specified, no prelude will be added.

quote => CODE|STRING
Mangle the lines of an `INLINE`d reply with CODE, or by prepending a STRING to each line. The routine specified by CODE is called when the line is in `$_`.

By default, `'> '` is added before each line. Specify `undef` to disable quoting. This option is processed after the body has been decoded.

signature => BODY|MESSAGE
The signature to be added in case of a multi-part reply. The mime-type of the signature body should indicate this is a used as such. However, in INLINE mode, the body will be taken, a line containing `'-- '`added before it, and added behind the epilogue.

strip_signature => REGEXP|STRING|CODE
Remove the signature of the sender. The value of this parameter is passed to Mail::Message::Body::stripSignature(pattern) unless the source text is not included. The signature is stripped from the message before quoting.

When a multipart body is encountered, and the message is included to ATTACH, the parts which look like signatures will be removed. If only one message remains, it will be the added as single attachment, otherwise a nested multipart will be the result. The value of this option does not matter, as long as it is present. See Mail::Message::Body::Multipart.

example:

```
    my $reply = $msg−>reply
     ( prelude         => "No spam, please!\n\n"
     , postlude        => "\nGreetings\n"
     , strip_signature => 1
     , signature       => $my_pgp_key
     , group_reply     => 1
     , 'X−Extra'       => 'additional header'
     );
```

$obj−>**replyPrelude**([STRING|$field|$address|ARRAY−$of−$things])

Produces a list of lines (usually only one), which will preceded the quoted body of the message. STRING must comply to the RFC822 email address specification, and is usually the content of a `To` or `From` header line. If a `$field` is specified, the field's body must be compliant. Without argument −or when the argument is `undef`− a slightly different line is produced.

An characteristic example of the output is

```
 On Thu Oct 13 04:54:34 1995, him@example.com wrote:
```

$obj−>**replySubject**(STRING)
Mail::Message−>**replySubject**(STRING)

Create a subject for a message which is a reply for this one. This routine tries to count the level of reply in subject field, and transform it into a standard form. Please contribute improvements.

example:

```
 subject                 −−> Re: subject
 Re: subject             −−> Re[2]: subject
 Re[X]: subject          −−> Re[X+1]: subject
 subject (Re)            −−> Re[2]: subject
 subject (Forw)          −−> Re[2]: subject
 <blank>                 −−> Re: your mail
```

DIAGNOSTICS

Error: Cannot include reply source as `$include`.

Unknown alternative for the `include` option of **reply()**. Valid choices are `NO`, `INLINE`, and `ATTACH`.

SEE ALSO

This module is part of Mail-Message distribution version 3.012, built on February 11, 2022. Website: *http://perl.overmeer.net/CPAN/*

LICENSE