

MailBox

German Perl Workshop
13 May 2025, München
Mark Overmeer (markov)

Once upon a time, I had to learn Perl

- In 1999, ...
- Mail / MIME
- MailTools

MailBox v3

- Mail-Box
- Mail-Message
- Mail-Transport
- Mail-Box-Parser-C
- Mail-Box-IMAP4
- Mail-Box-POP3
- MIME-Types
- User::Identity
- Mail-IMAPClient
- HTML-FromMail
- Mail-Box-Dbx
- Mail-Box-Netzwert
- OODoc

Mail-Box

- Folder Manager
 - manager
 - mbox, MH, Maildir, (IMAP4, POP3, ...)
 - locking
 - threads
 - search

```
use Mail::Box::Manager;
my $mgr = Mail::Box::Manager->new;

# Create folder objects.
my $folder = $mgr->open(folder => $ENV{MAIL});
my $message1 = $folder->message(0);
$mgr->copyMessage('Draft', $message);

my @messages = $folder->message(0,3);
$mgr->moveMessage('Outbox', @messages, create => 1);
$mgr->close($folder);

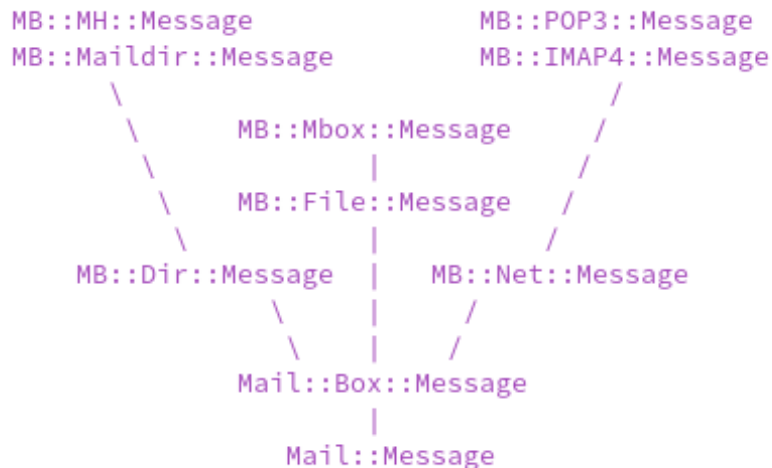
# Create thread-detectors (see Mail::Box::Thread::Manager)
my $t = $mgr->threads($inbox, $outbox);

my $threads = $mgr->threads(folder => $folder);
foreach my $thread ($threads->all)
{
    $thread->print;
}
```

Mail-Box folders

INHERITANCE

```
Mail::Box::MH::Message
  is a Mail::Box::Dir::Message
  is a Mail::Box::Message
  is a Mail::Message
  is a Mail::Reporter
```



Mail::Box::MH->new(%options)

-Option	--Defined in	--Default
access	Mail::Box	'r'
body_delayed_type	Mail::Box	Mail::Message::Body::Delayed
body_type	Mail::Box	Mail::Message::Body::Lines
coerce_options	Mail::Box	[]
create	Mail::Box	<false>
directory	Mail::Box::Dir	<derived from folder name>
extract	Mail::Box	10240
field_type	Mail::Box	undef
fix_headers	Mail::Box	<false>
folder	Mail::Box	\$ENV{MAIL}
folderdir	Mail::Box	\$ENV{HOME}/.mh
head_delayed_type	Mail::Box	Mail::Message::Head::Delayed
head_type	Mail::Box	Mail::Message::Head::Complete
index		undef
index_filename		<foldername>/index
index_type		Mail::Box::MH::Index
keep_dups	Mail::Box	<false>
keep_index		0
labels		undef
labels_filename		<foldername>/mh_sequence
labels_type		Mail::Box::MH::Labels
lock_file	Mail::Box	<index_file>
lock_timeout	Mail::Box	1 hour
lock_type	Mail::Box	Mail::Box::Locker::DotLock
lock_wait	Mail::Box	10 seconds
locker	Mail::Box	undef
log	Mail::Reporter	'WARNINGS'
manager	Mail::Box	undef
message_type	Mail::Box	<folder-class>::Message
multipart_type	Mail::Box	Mail::Message::Body::Multipart
remove_when_empty	Mail::Box	<true>
save_on_exit	Mail::Box	<true>
trace	Mail::Reporter	'WARNINGS'
trusted	Mail::Box	<depends on folder location>

```

Mail::Message::Head::Complete
|
Mail::Message isa
|
`---has a---> Mail::Message::Head::Partial
|
|      `-- has many --> MM::Field
|      `-- has many --> MM::Field::Full
|
|
`---has a---> Mail::Message::Body
| one of:
|      `--- MMB::Lines
|      `--- MMB::String
|      `--- MMB::File
|      `--- MMB::Nested
|      `--- one ---> MM::Message
|      `--- MMB::Multipart
|              `--- many --> MM::Part
|                      isa Mail::Message

```

Lots of classes

MailBox [Documentation](#)

3.011

produced 2025/04/18

Mail-Box

All Manuals

all manuals

[all methods](#)

[all diagnostics](#)

[all details](#)

[HTML::FromMail](#)
[HTML::FromMail::Field](#)
[HTML::FromMail::Format](#)
[HTML::FromMail::Format::Magic](#)
[HTML::FromMail::Format::OODoc](#)
[HTML::FromMail::Head](#)
[HTML::FromMail::Message](#)
[HTML::FromMail::Object](#)
[HTML::FromMail::Page](#)
[Html::FromMail::Default::HTMLifiers](#)
[Html::FromMail::Default::Previewers](#)
[MIME::Type](#)
[MIME::Types](#)
[Mail::Address](#)
[Mail::Box](#)
[Mail::Box::Cookbook](#)

[Mail::Box::Parser::C](#)
[Mail::Box::Parser::Perl](#)
[Mail::Box::Search](#)
[Mail::Box::Search::Grep](#)
[Mail::Box::Search::SpamAssassin](#)
[Mail::Box::Thread::Manager](#)
[Mail::Box::Thread::Node](#)
[Mail::Box::Tie](#)
[Mail::Box::Tie::ARRAY](#)
[Mail::Box::Tie::HASH](#)
[Mail::Cap](#)
[Mail::Field](#)
[Mail::Field::AddrList](#)
[Mail::Field::Date](#)
[Mail::Field::Generic](#)
[Mail::Filter](#)

[Mail::Message::Field::Full](#)
[Mail::Message::Field::Structured](#)
[Mail::Message::Field::URIs](#)
[Mail::Message::Field::Unstructured](#)
[Mail::Message::Head](#)
[Mail::Message::Head::Complete](#)
[Mail::Message::Head::Delayed](#)
[Mail::Message::Head::FieldGroup](#)
[Mail::Message::Head::ListGroup](#)
[Mail::Message::Head::Partial](#)
[Mail::Message::Head::ResentGroup](#)
[Mail::Message::Head::SpamGroup](#)
[Mail::Message::Head::Subset](#)
[Mail::Message::Part](#)
[Mail::Message::Replace::MailHeader](#)
[Mail::Message::Replace::MailInternet](#)

Lots of classes /2

Mail::Box-Cookbook
Mail::Box-Index
Mail::Box-Overview
Mail::Box::Collection
Mail::Box::Dbx
Mail::Box::Dbx::Message
Mail::Box::Dir
Mail::Box::Dir::Message
Mail::Box::FastScalar
Mail::Box::File
Mail::Box::File::Message
Mail::Box::IMAP4
Mail::Box::IMAP4::Head
Mail::Box::IMAP4::Message
Mail::Box::IMAP4s
Mail::Box::Identity
Mail::Box::Locker
Mail::Box::Locker::DotLock
Mail::Box::Locker::FcntlLock
Mail::Box::Locker::Flock
Mail::Box::Locker::Multi
Mail::Box::Locker::Mutt
Mail::Box::Locker::NFS
Mail::Box::Locker::POSIX

Mail::Filter
Mail::Header
Mail::Identity
Mail::Internet
Mail::Mailer
Mail::Message
Mail::Message::Body
Mail::Message::Body::Construct
Mail::Message::Body::Delayed
Mail::Message::Body::Encode
Mail::Message::Body::File
Mail::Message::Body::Lines
Mail::Message::Body::Multipart
Mail::Message::Body::Nested
Mail::Message::Body::String
Mail::Message::Construct
Mail::Message::Construct::Bounce
Mail::Message::Construct::Build
Mail::Message::Construct::Forward
Mail::Message::Construct::Read
Mail::Message::Construct::Rebuild
Mail::Message::Construct::Reply
Mail::Message::Construct::Text
Mail::Message::Convert

Mail::Message::Replace::MailInternet
Mail::Message::TransferEnc
Mail::Message::TransferEnc::Base64
Mail::Message::TransferEnc::Binary
Mail::Message::TransferEnc::EightBit
Mail::Message::TransferEnc::QuotedPrint
Mail::Message::TransferEnc::SevenBit
Mail::Message::Wrapper::SpamAssassin
Mail::Reporter
Mail::Send
Mail::Server
Mail::Server::IMAP4
Mail::Server::IMAP4::Fetch
Mail::Server::IMAP4::List
Mail::Server::IMAP4::Search
Mail::Server::IMAP4::User
Mail::Transport
Mail::Transport::Exim
Mail::Transport::IMAP4
Mail::Transport::Mailx
Mail::Transport::POP3
Mail::Transport::Qmail
Mail::Transport::Receive
Mail::Transport::SMTP

Lots of classes /3

Mail::Box::MH
Mail::Box::MH::Index
Mail::Box::MH::Labels
Mail::Box::MH::Message
Mail::Box::Maildir
Mail::Box::Maildir::Message
Mail::Box::Manage::User
Mail::Box::Manager
Mail::Box::Mbox
Mail::Box::Mbox::Message
Mail::Box::Message
Mail::Box::Message::Destructed
Mail::Box::Net
Mail::Box::Net::Message
Mail::Box::POP3
Mail::Box::POP3::Message
Mail::Box::POP3s
Mail::Box::Parser

Mail::Message::Convert::EmailSimple
Mail::Message::Convert::Html
Mail::Message::Convert::HtmlFormatPS
Mail::Message::Convert::HtmlFormatText
Mail::Message::Convert::MailInternet
Mail::Message::Convert::MimeEntity
Mail::Message::Convert::TextAutoformat
Mail::Message::Dummy
Mail::Message::Field
Mail::Message::Field::AddrGroup
Mail::Message::Field::Address
Mail::Message::Field::Addresses
Mail::Message::Field::Attribute
Mail::Message::Field::AuthResults
Mail::Message::Field::DKIM
Mail::Message::Field::Date
Mail::Message::Field::Fast
Mail::Message::Field::Flex

Mail::Transport::Send
Mail::Transport::Sendmail
Mail::Util
MailTools
MojoX::MIME::Types
Object::Realize::Later
User::Identity
User::Identity::Archive
User::Identity::Archive::Plain
User::Identity::Collection
User::Identity::Collection::Emails
User::Identity::Collection::Locations
User::Identity::Collection::Systems
User::Identity::Collection::Users
User::Identity::Item
User::Identity::Location
User::Identity::System

... These are only my own modules ...

Creating a Message

```
my $body = Mail::Message::Body::Lines->new
( file           => 'passphoto.jpg'
, mime_type      => 'image/jpeg'
, transfer_encoding => 'base64'
, disposition    => 'attachment'
);
```

```
my $msg = Mail::Message->buildFromBody
( $body
, From    => 'me@example.com'
, To      => 'you@example.com'
, Subject => 'This is me'
, ...other header lines...
);
```

```
$msg->send(via => 'postfix');
```

Creating a Message

```
Mail::Message->build  
( From      => 'me@example.com'  
  , To       => 'you@example.com'  
  , Subject  => 'This is me'  
  
  , data     => "Herewith my passphoto, as requested.\n"  
  , attach   => 'passphoto.jpg'  
  
)->send(via => 'postfix');
```

Also: read(), bounce(), forward(), rebuild(), and reply()

Does your library support this?

Subject: =?UTF-8?Q?This_Week!_-_Perl_Community_Conference,_Winter_2024_?=
=?UTF-8?Q?=F0=9F=8E=82-_Schedule_Posted?=

To: =?iso-8859-1?Q?S=F6ren_Laird_S=F6rries?= <soeren@soerries.de>

Mid-life upgrade

- 25 years old
 - new RFCs, still 100% compliant?
 - still using 5.8 / 5.10 syntax
 - Benchmarking
 - documentation production (OODoc)
 - error handling / exception framework
 - security evaluation

The logo for simpl=lists. It features the text "simpl=lists" in a dark teal, sans-serif font. The equals sign is stylized as a horizontal line with a small vertical bar in the center, resembling a list symbol.

Checking 96 RFCs

RFC2045, RFC2046, RFC2047, RFC2822, RFC3459, RFC3462, RFC3464, RFC3534, RFC3555, RFC3676, RFC3745, RFC3798, RFC3801, RFC3802, RFC3803, RFC3823, RFC3839, RFC3850, RFC3851, RFC3853, RFC3854, RFC3855, RFC3870, RFC3949, RFC3950, RFC4010, RFC4021, RFC4047, RFC4102, RFC4130, RFC4134, RFC4141, RFC4180, RFC4249, RFC4262, RFC4288, RFC4289, RFC4329, RFC4337, RFC4374, RFC4393, RFC4483, RFC4490, RFC4573, RFC4612, RFC5008, RFC5068, RFC5114, RFC5147, RFC5322, RFC5334, RFC5335, RFC5598, RFC5621, RFC5686, RFC5688, RFC5703, RFC5750, RFC5751, RFC5754, RFC5911, RFC5912, RFC6025, RFC6109, RFC6129, RFC6152, RFC6210, RFC6211, RFC6318, RFC6522, RFC6532, RFC6657, RFC6664, RFC6922, RFC7107, RFC7111, RFC7114, RFC7165, RFC7281, RFC7508, RFC7681, RFC7912, RFC8081, RFC8098, RFC8118, RFC8162, RFC8255, RFC8550, RFC8551, RFC8591, RFC8755, RFC8823, RFC9078, RFC9216, RFC9219, RFC9239

"New" Perl

- Mainly, replace use of `IO::Lines` and `Mail::Box::FastScalar`
- `s///r`
- `"use base" → "use parent"`
- ... etc

Improve documentation (OODoc)

=method clone %options

Create a copy of this message. Returned is a C<Mail::Message> object.
The head and body, the log and trace levels are taken. Labels are copied with the message, but the delete and modified flags are not.

See also **M<Mail::Box::Message::copyTo()>** and M<Mail::Box::Message::moveTo()>.

=option shallow_head BOOLEAN

=default shallow_head <false>

Only the head uses is reused, not the body. This is probably a bad choice, because the header fields can be updated, for instance when labels change.

=option shallow_body BOOLEAN

=default shallow_body <false>

A rather safe bet, because you are not allowed to modify the body of a message: you may only set a new body with M<body()>.

=example

```
$copy = $msg->clone;
```

=cut

OODoc in POD

=item \$obj-E<gt>B<clone>(%options)

Create a copy of this message. [...]

See also L<Mail::Box::Message::copyTo()|Mail::Box::Message/"The message"> and L<Mail::Box::Message::moveTo()|Mail::Box::Message/"The message">.

-Option --Default
shallow_body <false>
shallow_head <false>

=over 2

=item shallow_body => BOOLEAN

A rather safe bet, because you are not allowed to modify the body of a message: you may only set a new body with L<body()|Mail::Message/"The body">.

=item shallow_head => BOOLEAN

Only the head uses is reused, not the body. This is probably a bad choice, because the header fields can be updated, for instance when labels change.

=back

example:

```
$copy = $msg->clone;
```

OODoc in man

`$obj->clone(%options)`

Create a copy of this message. Returned is a "Mail::Message" object. The head and body, the log and trace levels are taken. Labels are copied with the message, but the delete and modified flags are not.

See also **Mail::Box::Message::copyTo()** and **Mail::Box::Message::moveTo()**.

-Option --Default
 shallow_body <false>
 shallow_head <false>

shallow_body => BOOLEAN

A rather safe bet, because you are not allowed to modify the body of a message: you may only set a new body with **body()**.

shallow_head => BOOLEAN

Only the head uses is reused, not the body. This is probably a bad choice, because the header fields can be updated, for instance when labels change.

example:

```
$copy = $msg->clone;
```

OODoc in HTML

`$obj->clone( %options )`

Create a copy of this message. Returned is a `Mail::Message` object. The head and body, the log and trace levels are taken. Labels are copied with the message, but the delete and modified flags are not.

See also `Mail::Box::Message::copyTo()` and `Mail::Box::Message::moveTo()`.

Option	Default
<code>shallow_body</code>	<code><false></code>
<code>shallow_head</code>	<code><false></code>

`shallow_body` => BOOLEAN

A rather safe bet, because you are not allowed to modify the body of a message: you may only set a new body with `body()`.

`shallow_head` => BOOLEAN

Only the head uses is reused, not the body. This is probably a bad choice, because the header fields can be updated, for instance when labels change.

» Example:

```
$copy = $msg->clone;
```

Mail::Message — Mozilla Firefox

Stati

Lives

Stati

Stati

(19) F

Gern

Mast

WHY

FOSI

Mail:

Gern

Refel

NPO PLAYI

M: x

← → ↺

https://perl.overmeer.net/mailbox/html/Mail_Message/index.html

☆

📧 ⬇️ 👤 📁 ☰

MailBox documentation

Mail-Message 3.018

produced 2025/04/18

Mail::Message

general message object

[all manuals](#)

[all methods](#)

[all diagnostics](#)

[all details](#)

SYNOPSIS

```
use Mail::Box::Manager;
my $mgr = Mail::Box::Manager->new;
my $folder = $mgr->open(folder => 'InBox');
my $msg = $folder->message(2); # $msg is a Mail::Message now

my $subject = $msg->subject; # The message's subject
my @cc = $msg->cc; # List of Mail::Address'es

my Mail::Message::Head $head = $msg->head;
my Mail::Message::Body $body = $msg->decoded;
$msg->decoded->print($outfile);

# Send a simple email
Mail::Message->build
( To => 'you@example.com'
, From => 'me@example.com'
, Subject => "My subject"
, data => "Some plain text content"
)->send(via => 'postfix');

my $reply_msg = Mail::Message->reply(...);
my $frwd_msg = Mail::Message->forward(...);
```

See [SYNOPSIS](#) in [Mail::Reporter](#)

DESCRIPTION

A `Mail::Message` object is a container for MIME-encoded message information, as defined by RFC2822. Everything what is not specifically related to storing the messages in mailboxes (folders) is implemented in this class. Methods which are related to folders is implemented in the [Mail::Box::Message](#) extension.

Show

- [class relations](#)
- methods [grouped](#)
- methods [alphabetically](#)
- [documentation overview](#)

INHERITANCE

Mail::Message has extra code in

- [Mail::Message::Construct](#)
- [Mail::Message::Construct::Bounce](#)
- [Mail::Message::Construct::Build](#)
- [Mail::Message::Construct::Forward](#)
- [Mail::Message::Construct::Read](#)
- [Mail::Message::Construct::Rebuild](#)
- [Mail::Message::Construct::Reply](#)
- [Mail::Message::Construct::Text](#)

Mail::Message is a [Mail::Reporter](#)

Mail::Message is extended by

- [Mail::Box::Message](#)
- [Mail::Message::Dummy](#)
- [Mail::Message::Part](#)
- [Mail::Message::Replace::MailInternet](#)

Exception handling

- Mail::Reporter is generic base-class

- 482 errors
- 176 warnings

```
$folder->log(WARNING => 'go away');  
print $folder->trace;           # current level  
$folder->trace('PROGRESS');    # set level  
print $folder->errors;  
print $folder->report('PROGRESS');
```

- → Log::Report

- MailBox v4

```
print $message->log;           # may print "NOTICE"  
print $message->log +0;        # may print "3"  
$message->log('ERRORS');      # sets a new level, returns the numeric value  
  
$message->log(WARNING => "This message is too large.");  
$folder ->log(NOTICE => "Cannot read from file $filename.");  
$manager->log(DEBUG => "Hi there!", reverse sort @l);  
  
Mail::Message->log(ERROR => 'Unknown');
```

MailBox

mid-life