NAME

Mail::Message::Head::ListGroup − mailinglist related header fields

INHERITANCE

```
 Mail::Message::Head::ListGroup
   is a Mail::Message::Head::FieldGroup
   is a Mail::Reporter
```

SYNOPSIS

```
 my $lg = Mail::Message::Head::ListGroup−>new(head => $head, ...);
 $head−>addListGroup($lg);

 my $lg = $head−>addListGroup(...);

 $lg−>delete;
```

DESCRIPTION

A *list group* is a set of header fields which are added by mailing-list managing software. This class knowns various details about that software.

The knowledge and test messages which are used to initially implement this module is taken from Mail::ListDetector, written by Michael Stevens <mailto:michael@etla.org>. The logic is redesigned to add flexibility and use the powerful MailBox features.

Extends "DESCRIPTION" in Mail::Message::Head::FieldGroup.

METHODS

Extends "METHODS" in Mail::Message::Head::FieldGroup.

Constructors

Extends "Constructors" in Mail::Message::Head::FieldGroup.

`$obj`−>**address**()

Returns a Mail::Message::Field::Address object (or `undef`) which defines the posting address of the mailing list.

`$obj`−>**clone**()

Inherited, see "Constructors" in Mail::Message::Head::FieldGroup

`$obj`−>**from**($head|$message)

Create a `Mail::Message::Head::ListGroup` based in the specified `$message` or message `$head`.

`$obj`−>**implementedTypes**()
Mail::Message::Head::ListGroup−>**implementedTypes**()

Inherited, see "Constructors" in Mail::Message::Head::FieldGroup

`$obj`−>**listname**()

Returns the name of the mailing list, which is usually a part of the e−mail address which is used to post the messages to.

Mail::Message::Head::ListGroup−>**new**($fields, `%options`)

Construct an object which maintains one set of mailing list headers

```
 −Option  −−Defined in                     −−Default
  address                                    undef
  head      Mail::Message::Head::FieldGroup  undef
  listname                                   <derived from address>
  log       Mail::Reporter                   'WARNINGS'
  rfc                                        undef
  software  Mail::Message::Head::FieldGroup  undef
  trace     Mail::Reporter                   'WARNINGS'
  type      Mail::Message::Head::FieldGroup  undef
  version   Mail::Message::Head::FieldGroup  undef
```

address => STRING|OBJECT

Address of the mailing list, which may be specified as STRING or e−mail containing object (a Mail::Address or Mail::Identity. In any case, the data is converted into a Mail::Identity.

head => HEAD

listname => STRING

A short textual representation of the mailing-list.

log => LEVEL

rfc => ’rfc2919’|’rfc2369’

Defines the mailing list software follows an rfc.

software => STRING

trace => LEVEL

type => STRING

version => STRING

`$obj`−>**rfc**()

When the mailing list software follows the guidelines of one of the dedicated RFCs, then this will be returned otherwise `undef`. The return values can be `rfc2919`, `rfc2369`, or `undef`.

The header

Extends ‘‘The header’’ in Mail::Message::Head::FieldGroup.

`$obj`−>**add**(<$field, `$value`> | `$object`)

Inherited, see ‘‘The header’’ in Mail::Message::Head::FieldGroup

`$obj`−>**addFields**([$fieldnames])

Inherited, see ‘‘The header’’ in Mail::Message::Head::FieldGroup

`$obj`−>**attach**($head)

Inherited, see ‘‘The header’’ in Mail::Message::Head::FieldGroup

`$obj`−>**delete**()

Inherited, see ‘‘The header’’ in Mail::Message::Head::FieldGroup

`$obj`−>**fieldNames**()

Inherited, see ‘‘The header’’ in Mail::Message::Head::FieldGroup

`$obj`−>**fields**()

Inherited, see ‘‘The header’’ in Mail::Message::Head::FieldGroup

`$obj`−>**head**()

Inherited, see ‘‘The header’’ in Mail::Message::Head::FieldGroup

Access to the header

Extends ‘‘Access to the header’’ in Mail::Message::Head::FieldGroup.

`$obj`−>**isListGroupFieldName**($name)

Mail::Message::Head::ListGroup−>**isListGroupFieldName**($name)

`$obj`−>**software**()

Inherited, see ‘‘Access to the header’’ in Mail::Message::Head::FieldGroup

$obj−>**type**()
Inherited, see ‘‘Access to the header’’ in Mail::Message::Head::FieldGroup

$obj−>**version**()
Inherited, see ‘‘Access to the header’’ in Mail::Message::Head::FieldGroup

Internals

Extends ‘‘Internals’’ in Mail::Message::Head::FieldGroup.

$obj−>**collectFields**()
Scan the header for fields which are usually contained in mailing list software. This method is automatically called when a list group is constructed **from()** an existing header or message.

Returned are the names of the list header fields found, in scalar context the amount. An empty list/zero indicates that this is not a mailing list message.

Please warn the author of MailBox if you see that to few or too many fields are included.

$obj−>**detected**($type, $software, $version)
Inherited, see ‘‘Internals’’ in Mail::Message::Head::FieldGroup

Error handling

Extends ‘‘Error handling’’ in Mail::Message::Head::FieldGroup.

$obj−>**AUTOLOAD**()
Inherited, see ‘‘Error handling’’ in Mail::Reporter

$obj−>**addReport**($object)
Inherited, see ‘‘Error handling’’ in Mail::Reporter

$obj−>**defaultTrace**([$level]|[$loglevel, $tracelevel]|[$level, $callback])
Mail::Message::Head::ListGroup−>**defaultTrace**([$level]|[$loglevel, $tracelevel]|[$level, $callback])
Inherited, see ‘‘Error handling’’ in Mail::Reporter

$obj−>**details**()
Produce information about the detected/create list group, which may be helpful during debugging, by default to the selected file handle.

$obj−>**errors**()
Inherited, see ‘‘Error handling’’ in Mail::Reporter

$obj−>**log**([$level, [$strings]])
Mail::Message::Head::ListGroup−>**log**([$level, [$strings]])
Inherited, see ‘‘Error handling’’ in Mail::Reporter

$obj−>**logPriority**($level)
Mail::Message::Head::ListGroup−>**logPriority**($level)
Inherited, see ‘‘Error handling’’ in Mail::Reporter

$obj−>**logSettings**()
Inherited, see ‘‘Error handling’’ in Mail::Reporter

$obj−>**notImplemented**()
Inherited, see ‘‘Error handling’’ in Mail::Reporter

$obj−>**print**([$fh])
Inherited, see ‘‘Error handling’’ in Mail::Message::Head::FieldGroup

$obj−>**report**([$level])
Inherited, see ‘‘Error handling’’ in Mail::Reporter

$obj−>**reportAll**([$level])
Inherited, see ‘‘Error handling’’ in Mail::Reporter

`$obj->`**trace**([$level])
Inherited, see ‘‘Error handling’’ in Mail::Reporter

`$obj->`**warnings**()
Inherited, see ‘‘Error handling’’ in Mail::Reporter

Cleanup

Extends ‘‘Cleanup’’ in Mail::Message::Head::FieldGroup.

`$obj->`**DESTROY**()
Inherited, see ‘‘Cleanup’’ in Mail::Reporter

DETAILS

Mailing list fields

Detected lists

The Mail::Message::Head::ListGroup class can detect many different mailing lists, some of which are very popular and some of which are rare.

Numerous fields in a header are added when the message is passed through a mailing list server. Each list software has defined its own fields, sometimes woth conflicting definitions. There are also two RFCs about mailing list: `rfc2919` and `rfc2369`.

The following lists are currently detected. Between parenthesis is the string returned by **type()** when that differs from the software name.

- CommuniGate

  Legacy commercial MacOS implementation by Stalker Software Inc. <http://www.stalker.com/mac/default.html>

- CommuniGate Pro (CommuniGatePro)

  Commercial rfc2919 compliant implementation by Stalker Software Inc. <http://www.stalker.com>

- Ecartis

  Commercial mailing list manager, formerly known as Listar. Produced by NodeRunner Computing. See <http://www.ecartis.com>.

- Ezmlm

  Open Source mailing list manager, available from <http://www.ezmlm.org>.

- FML

  Open Source mailing list manager, see <http://www.fml.org>.

- Listar

  Old name for Ecartis.

- Listbox

  Mailing lists defined at <http://listbox.com>.

- Mailman

  GNU’s mailing list manager, available from <http://www.list.org>.

- Majordomo

  Free (licenced) mailing list manager by Great Circle Associates, available from <http://www.greatcircle.com/majordomo/>

- Smartlist

  Related to procmail, as expressed by their shared main page at <http://www.procmail.org/>.

- Yahoo! Groups (YahooGroups)

  Mailing lists defined at <http://groups.yahoo.com>.

- Listserv

  Commercial mailing list manager, produced by L−Soft. See <http://www.lsoft.com/>.

DIAGNOSTICS

Error: Cannot convert ‘‘$string’’ into an address object

The new(address) is coerced into a Mail::Message::Field::Address, which fails. Have a look at **Mail::Message::Field::Address::coerce()** to see what valid arguments are.

Error: Package `$package` does not implement `$method`.

Fatal error: the specific package (or one of its superclasses) does not implement this method where it should. This message means that some other related classes do implement this method however the class at hand does not. Probably you should investigate this and probably inform the author of the package.

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