






- List::distribute\_msg() for distribution (see [list-distribute-msg](#), page )
- List::send\_auth() for authentication or topic tagging by message sender(see [list-send-auth](#), page )
- List::send\_to\_editor() for moderation or topic tagging by list moderator(see [list-send-to-editor](#), page )
- List::automatic\_tag() for automatic topic tagging (see [list-automatic-tag](#), page )

**IN:**

1. which(+): 'list\_name@domain\_name - the concerned list
2. message(+): ref(Message) - sent message
3. robot(+): robot

**OUT:** 1 *if everything went fine* in order to remove the file from the queue undef

**DoCommand()**


Handles a command sent to sympa. The command is parse by calling Command::parse() (see [commands-parse](#), page )

**IN:**

1. rcpt: receipient <listname>-<subscribeunsubscribe>
2. robot(+): robot
3. msg(+): ref(MIME::Entity) - message containing the command
4. file(+): file containing the message

**OUT:** \$success - result of Command::parse() function undef.

**DoSendMessage()**


Sends a message pushed in spool by another process (ex: wwsympa.fcgi) by calling function mail::mail\_forward() (see [mail-mail-forward](#), page )

**IN:**

1. msg(+): ref(MIME::Entity)
2. robot(+): robot

**OUT:** 1 undef

**DoForward()**


Handles a message sent to <listname>-editor: the list editor, <list>-request: the list owner or the listmaster. The message is forwarded according to \$function by calling function mail::mail\_forward() (see [mail-mail-forward](#), page )

**IN:**

1. name(+): list name *if (\$function != 'listmaster')*
2. function(+): 'listmaster' 'request' 'editor'
3. robot(+): robot
4. msg(+): ref(MIME::Entity)

**OUT:** 1 undef

## SendDigest()

Reads the queuedigest spool and send old digests to the subscribers with the digest option by calling List::send\_msg\_digest() function mail::mail\_forward() (see [list-send-msg-digest](#), page )

**IN:** - **OUT:** - undef

## CleanSpool()


Cleans old files from spool \$spool\_dir older than \$clean\_delay.

**IN:**

1. spool\_dir(+): the spool directory
2. clean\_delay(+): the delay in days


**OUT:** 1

## sigterm()

This function is called when a signal -TERM is received by sympa.pl. It just changes the value of \$signal loop variable in order to stop sympa.pl after ending its message distribution if in progress. (see [stop-signals](#), page )

**IN:** - **OUT:** -

## sighup()

This function is called when a signal -HUP is received by sympa.pl. It changes the value of \$signal loop variable and switches of the -mail (see [stop-signals](#), page ) logging option and continues current task.

**IN:** - **OUT:** -


# Commands.pm

This module does the mail commands processing.

## Commands processing

parse(), add(), del(), subscribe(), signoff(), invite(), last(), index(), getfile(), confirm(), set(), distribute(), reject(), modindex(), review(), verify(), remind(), info(), stats(), help(), lists(), which(), finished().

### parse()

Parses the command line and calls the adequate subroutine (following functions) with the arguments of the command. This function is called by `sympa::DoCommand()` (see [sympa-docommand](#), page ).

#### IN:

1. `sender(+)`: the command sender
2. `robot(+)`: robot
3. `i(+)`: command line
4. `sign_mod`: 'smime' undef

**OUT:** 'unknown\_cmd' \$status - command process result

### add()

Adds a user to a list (requested by another user), and can send acknowledgements. New subscriber can be notified by sending template 'welcome'.

#### IN:

1. `what(+)`: command parameters: listname, email and comments eventually
2. `robot(+)`: robot
3. `sign_mod`: 'smime' undef - authentication

**OUT:** 'unknown\_list' 'wrong\_auth' 'not\_allowed' 1 undef

### del()

Removes a user to a list (requested by another user), and can send acknowledgements. Unsubscriber can be notified by sending template 'removed'.

#### IN:

1. `what(+)`: command parameters: listname and email
2. `robot(+)`: robot
3. `sign_mod`: 'smime' undef - authentication

**OUT:** 'unknown\_list' 'wrong\_auth' 'not\_allowed' 1

## subscribe()

Subscribes a user to a list. New subscriber can be notified by sending him template 'welcome'.

### IN:

1. what(+): command parameters: listname and comments eventually
2. robot(+): robot
3. sign\_mod: 'smime' undef - authentication

**OUT:** 'unknown\_list' 'wrong\_auth' 'not\_allowed' 1 undef

## signoff()

Unsubscribes a user from a list. He can be notified by sending him template 'bye'.

### IN:

1. which(+): command parameters: listname and email
2. robot(+): robot
3. sign\_mod: 'smime' undef - authentication

**OUT:** 'syntax\_error' 'unknown\_list' 'wrong\_auth' 'not\_allowed' 1 undef

## invite()

Invites someone to subscribe to a list by sending him the template 'invite'.

### IN:

1. what(+): command parameters: listname, email and comments
2. robot(+): robot
3. sign\_mod: 'smime' undef - authentication

**OUT:** 'unknown\_list' 'wrong\_auth' 'not\_allowed' 1 undef

## last()

Sends back the last archive file by calling List::archive\_send() function (see [list-archive-send](#), page ).

### IN:

1. which(+): listname
2. robot(+): robot

**OUT:** 'unknown\_list' 'no\_archive' 'not\_allowed' 1

## index()


Sends the list of archived files of a list.

### IN:

1. which(+): listname
2. robot(+): robot

**OUT:** 'unknown\_list' 'no\_archive' 'not\_allowed' 1

## getfile()


Sends back the requested archive file by calling List::archive\_send() function (see [list-archive-send](#), page )

### IN:

1. which(+): commands parameters: listname and filename(archive file)
2. robot(+): robot

**OUT:** 'unknown\_list' 'no\_archive' 'not\_allowed' 1

## confirm()

Confirms the authentication of a message for its distribution on a list by calling function List::distribute\_msg() for distribution (see [list-distribute-msg](#)) or by calling List::send\_to\_editor() for moderation (see )

### IN:

1. what(+): authentication key (command parameter)
2. robot(+): robot

**OUT:** 'wrong\_auth' 'msg\_not\_found' 1 undef

## set()

Changes subscription options (reception or visibility)

### IN:

1. what(+): command parameters: listname and reception mode (digestdigestplainnomailnormal...) or visibility mode(concealnoconceal).
2. robot(+): robot

**OUT:** 'syntax\_error' 'unknown\_list' 'not\_allowed' 'failed' 1

## distribute()

Distributes the broadcast of a validated moderated message.

### IN:

1. what(+): command parameters: listname and authentication key
2. robot(+): robot

**OUT:** 'unknown\_list' 'msg\_not\_found' 1 undef

## reject()

Refuses and deletes a moderated message. Rejected message sender can be notified by sending him template 'reject'.

### IN:

1. what(+): command parameters: listname and authentication key
2. robot(+): robot

**OUT:** 'unknown\_list' 'wrong\_auth' 1 undef

From:

<https://www.sympa.org/> - **Sympa mailing list server**

Permanent link:

[https://www.sympa.org/internals/internals\\_listpm3](https://www.sympa.org/internals/internals_listpm3)



Last update: **2006/12/18 15:26**