

NAME

Mail::Box::Collection – a collection of subfolders

INHERITANCE

```
Mail::Box::Collection
  is an User::Identity::Collection
  is an User::Identity::Item
```

```
Mail::Box::Collection
  is a Mail::Reporter
```

SYNOPSIS**DESCRIPTION**

The Mail::Box::Collection object maintains a set Mail::Box::Identity objects, each describing a the location of a single mailbox (folder). The collection is used by the Mail::Box::Manage::User object to administer the folders of a single user, even if those folders are closed.

Extends “DESCRIPTION” in Mail::Reporter.

Extends “DESCRIPTION” in User::Identity::Collection.

OVERLOADED

Extends “OVERLOADED” in User::Identity::Collection.

overload: @{}
 Inherited, see “OVERLOADED” in User::Identity::Collection

overload: **stringification**
 Inherited, see “OVERLOADED” in User::Identity::Collection

METHODS

Extends “METHODS” in Mail::Reporter.

Extends “METHODS” in User::Identity::Collection.

Constructors

Extends “Constructors” in Mail::Reporter.

Extends “Constructors” in User::Identity::Collection.

Mail::Box::Collection->**new**([\$name], %options)

--Option	--Defined in	--Default
description	User::Identity::Item	undef
folder_type		<from parent>
item_type	User::Identity::Collection	Mail::Box::Identity
log	Mail::Reporter	'WARNINGS'
manager		<from parent>
name	User::Identity::Item	'folders'
parent	User::Identity::Item	undef
roles	User::Identity::Collection	undef
trace	Mail::Reporter	'WARNINGS'

description => STRING

folder_type => CLASS

item_type => CLASS

log => LEVEL

manager => OBJECT

An Mail::Box::Manager OBJECT (could be a Mail::Box::Manage::User).

name => STRING

parent => OBJECT

```
roles => ROLE|ARRAY
trace => LEVEL
```

Error handling

Extends “Error handling” in Mail::Reporter.

`$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::Box::Collection->**defaultTrace**([\$level][\$loglevel, \$tracelevel][\$level, \$callback])

Inherited, see “Error handling” in Mail::Reporter

`$obj->errors()`

Inherited, see “Error handling” in Mail::Reporter

`$obj->log([$level, [$strings]])`

Mail::Box::Collection->**log**([\$level, [\$strings]])

Inherited, see “Error handling” in Mail::Reporter

`$obj->logPriority($level)`

Mail::Box::Collection->**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

`$obj->report([$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::Reporter.

`$obj->DESTROY()`

Inherited, see “Cleanup” in Mail::Reporter

Attributes

Extends “Attributes” in User::Identity::Collection.

`$obj->description()`

Inherited, see “Attributes” in User::Identity::Item

`$obj->folderType([$folderclass])`

Returns the type of folder (on this location). When specified, then `$folderclass` must be a Mail::Box extension.

`$obj->itemType()`

Inherited, see “Attributes” in User::Identity::Collection

`$obj->manager()`

The Mail::Box::Manager (usually a Mail::Box::Manage::User object), which is handling the opening of folders.

`$obj->name([$newname])`

Inherited, see “Attributes” in User::Identity::Item

`$obj->roles()`

Inherited, see “Attributes” in User::Identity::Collection

Collections

Extends “Collections” in User::Identity::Collection.

`$obj->add($collection, $role)`

Inherited, see “Collections” in User::Identity::Item

`$obj->addCollection($object | <[$type], %options>)`

Inherited, see “Collections” in User::Identity::Item

`$obj->collection($name)`

Inherited, see “Collections” in User::Identity::Item

`$obj->parent([$parent])`

Inherited, see “Collections” in User::Identity::Item

`$obj->removeCollection($object|$name)`

Inherited, see “Collections” in User::Identity::Item

`$obj->type()`

Mail::Box::Collection->type()

Inherited, see “Collections” in User::Identity::Item

`$obj->user()`

Inherited, see “Collections” in User::Identity::Item

Maintaining roles

Extends “Maintaining roles” in User::Identity::Collection.

`$obj->addRole($role| <[$name],%options> | ARRAY)`

Inherited, see “Maintaining roles” in User::Identity::Collection

`$obj->removeRole($role|$name)`

Inherited, see “Maintaining roles” in User::Identity::Collection

`$obj->renameRole(<$role|$oldname>, $newname)`

Inherited, see “Maintaining roles” in User::Identity::Collection

`$obj->sorted()`

Inherited, see “Maintaining roles” in User::Identity::Collection

Searching

Extends “Searching” in User::Identity::Collection.

`$obj->find($name|CODE|undef)`

Inherited, see “Searching” in User::Identity::Collection

DIAGNOSTICS

Error: `$object` is not a collection.

The first argument is an object, but not of a class which extends User::Identity::Collection.

Error: Cannot create a `$type` to add this to my collection.

Some options are specified to create a `$type` object, which is native to this collection. However, for some reason this failed.

Error: Cannot load collection module for `$type` (`$class`).

Either the specified `$type` does not exist, or that module named `$class` returns compilation errors. If the type as specified in the warning is not the name of a package, you specified a nickname which was not defined. Maybe you forgot the 'require' the package which defines the nickname.

Error: Cannot rename `$name` into `$newname`: already exists

Error: Cannot rename `$name` into `$newname`: doesn't exist

Error: Creation of a collection via `$class` failed.

The `$class` did compile, but it was not possible to create an object of that class using the options you specified.

Error: Don't know what type of collection you want to add.

If you add a collection, it must either be a collection object or a list of options which can be used to create a collection object. In the latter case, the type of collection must be specified.

Warning: No collection `$name`

The collection with `$name` does not exist and can not be created.

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.

Error: Wrong type of role for `$collection`: requires a `$expect` but got a `$type`

Each `$collection` groups sets of roles of one specific type (`$expect`). You cannot add objects of a different `$type`.

SEE ALSO

This module is part of Mail-Box distribution version 3.009, built on August 18, 2020. Website: <http://perl.overmeer.net/CPAN/>

LICENSE

Copyrights 2001–2020 by [Mark Overmeer]. For other contributors see ChangeLog.

This program is free software; you can redistribute it and/or modify it under the same terms as Perl itself. See <http://dev.perl.org/licenses/>