

**NAME**

User::Identity::Archive – base class for archiving user information

**INHERITANCE**

User::Identity::Archive  
is a User::Identity::Item

User::Identity::Archive is extended by  
User::Identity::Archive::Plain

**SYNOPSIS**

```
use User::Identity::Archive::Plain;
my $friends = User::Identity::Archive::Plain->new('friends');
$friends->from(\*FH);
$friends->from('.friends');
```

**DESCRIPTION**

An archive stores collections. It depends on the type of archive how and where that is done. Some archivers may limit the kinds of selections which can be stored.

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

**OVERLOADED****METHODS**

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

**Constructors**

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

```
User::Identity::Archive->new( [$name], %options )
--Option      --Defined in      --Default
description  User::Identity::Item  undef
from         User::Identity::Item  undef
name         User::Identity::Item  <required>
parent       User::Identity::Item  undef

description => STRING
from => FILEHANDLE|FILENAME
name => STRING
parent => OBJECT
```

**Attributes**

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

\$obj->**description**()  
Inherited, see “Attributes” in User::Identity::Item

\$obj->**name**( [\$newname] )  
Inherited, see “Attributes” in User::Identity::Item

**Collections**

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

\$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()`  
 User::Identity::Archive->type()  
 Inherited, see “Collections” in User::Identity::Item

`$obj->user()`  
 Inherited, see “Collections” in User::Identity::Item

### Searching

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

`$obj->find($collection, $role)`  
 Inherited, see “Searching” in User::Identity::Item

### Access to the archive

`$obj->from($source, %options)`  
 Read definitions from the specified `$source`, which usually can be a filehandle or filename. The syntax used in the information `$source` is archiver dependent.

Not all archivers implement `from()`, so you may want to check with `UNIVERSAL::can()` beforehand.

example:

```
use User::Identity::Archive::Some;
my $a = User::Identity::Archive::Some->new('xyz');
$a->from(*STDIN) if $a->can('from');
```

## 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 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: 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.

## SEE ALSO

This module is part of User-Identity distribution version 1.01, built on February 11, 2022. Website: <http://perl.overmeer.net/CPAN/>

## LICENSE

Copyrights 2003–2022 by [Mark Overmeer <markov@cpan.org>]. 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/>