

**NAME**

Lintian::Unpack::Task — Tasks when unpacking collections

**SYNOPSIS**

```
use Lintian::Unpack::Task;

my $task = Lintian::Unpack::Task->new;
```

**DESCRIPTION**

A task class for unpacking lab entries.

**CLASS METHODS**

`new (COLLMAP, PROFILE[, OPTIONS])`  
Creates a new unpacker.

`COLLMAP` is a `Lintian::DepMap::Properties` describing the dependencies between the collections. Each node in `COLLMAP` must have a `Lintian::CollScript` as property.

`OPTIONS` is an optional hashref containing optional configurations. If a key is not present, its value is assumed to be `undef` unless otherwise stated. The following key/values are available:

**“profile”**

If this key is present and its value is defined, the value must be `Lintian::Profile`. The unpacker will use the enabled checks of the Profile to determine what collections to use.

If “profile” is not present or its value is undefined, then all collections in `COLLMAP` will be unpacked.

**“extra-coll”**

If this key is present and its value is defined, it must be a reference to a hash table. The keys are considered names of “extra” collections to unpack. The values in this table is ignored.

Extra collections will be unpacked on top of other collections.

NB: This value is ignored if “profile” is not given.

**“jobs”**

This value is the max number of jobs to be run in parallel. Can be changed with the “jobs” method later. If omitted, it defaults to 0. Refer to “jobs” for more info.

**INSTANCE METHODS**

`prepare_tasks (ERRHANDLER, LAB-ENTRY...)`  
Prepare a number of lab entries for unpacking.

The `ERRHANDLER` should be a code ref, which will be invoked in case that an entry is not in the laboratory and cannot be created (via the `create` method). It is invoked once per failed entry giving the entry as first (and only) argument.

If `ERRHANDLER` returns normally, the entry is skipped (and will not be unpacked later). If `ERRHANDLER` croaks/dies/etc., the method will attempt to update the status file for any entry it created before passing back the error to the caller (via `die`).

`LAB-ENTRY` is an array of lab entries to be processed. They must be instances of `Lintian::Lab::Entry`, but do not have to exist. They will be created as needed.

Returns a truth value if at least one entry needs to be processed and it did not cause an error. Otherwise, it returns `undef`.

NB: The status file is not updated for created entries on successful return. It should either be done by running the `process_tasks` method or manually.

**AUTHOR**

Originally written by Felix Lechner <felix.lechner@lease-up.com> for Lintian.

## SEE ALSO

**lintian** (1), **Lintian::CollScript** (3), **Lintian::Lab::Entry** (3)