NAME

CPANPLUS::Dist - base class for plugins

SYNOPSIS

```
my $dist = CPANPLUS::Dist::YOUR_DIST_TYPE_HERE->new(
                            module  => $modobj,
                    );
```

DESCRIPTION

CPANPLUS::Dist is a base class for CPANPLUS::Dist::MM and CPANPLUS::Dist::Build. Developers of other CPANPLUS::Dist::* plugins should look at CPANPLUS::Dist::Base.

ACCESSORS

parent()

> Returns the CPANPLUS::Module object that parented this object.

status()

> Returns the Object::Accessor object that keeps the status for this module.

STATUS ACCESSORS

All accessors can be accessed as follows: $deb->status->ACCESSOR

created()

> Boolean indicating whether the dist was created successfully. Explicitly set to 0 when failed, so a value of undef may be interpreted as not yet attempted.

installed()

> Boolean indicating whether the dist was installed successfully. Explicitly set to 0 when failed, so a value of undef may be interpreted as not yet attempted.

uninstalled()

> Boolean indicating whether the dist was uninstalled successfully. Explicitly set to 0 when failed, so a value of undef may be interpreted as not yet attempted.

dist()

> The location of the final distribution. This may be a file or directory, depending on how your distribution plug in of choice works. This will be set upon a successful create.

$dist = CPANPLUS::Dist::YOUR_DIST_TYPE_HERE->new(module => MODOBJ);

Create a new CPANPLUS::Dist::YOUR_DIST_TYPE_HERE object based on the provided MODOBJ.

*** DEPRECATED *** The optional argument format is used to indicate what type of dist you would like to create (like CPANPLUS::Dist::MM or CPANPLUS::Dist::Build and so on).

CPANPLUS::Dist->new is exclusively meant as a method to be inherited by CPANPLUS::Dist::MM|Build.

Returns a CPANPLUS::Dist::YOUR_DIST_TYPE_HERE object on success and false on failure.

@dists = CPANPLUS::Dist->dist_types;

Returns a list of the CPANPLUS::Dist::* classes available

$bool = CPANPLUS::Dist->rescan_dist_types;

Rescans @INC for available dist types. Useful if you've installed new CPANPLUS::Dist::* classes and want to make them available to the current process.

$bool = CPANPLUS::Dist->has_dist_type($type)

Returns true if distribution type $type is loaded/supported.

$bool = $dist->prereq_satisfied(modobj => $modobj, version => $version_spec)

Returns true if this prereq is satisfied. Returns false if it's not. Also issues an error if it seems "unsatisfiable," i.e. if it can't be found on CPAN or the latest CPAN version doesn't satisfy it.

$configure_requires = $dist->find_configure_requires([file => /path/to/META.yml])

Reads the configure_requires for this distribution from the META.yml or META.json file in the root directory and returns a hashref with module names and versions required.

$bool = $dist->_resolve_prereqs(...)

Makes sure prerequisites are resolved

```
format          The dist class to use to make the prereqs
                (ie. CPANPLUS::Dist::MM)


prereqs         Hash of the prerequisite modules and their versions


target          What to do with the prereqs.
                    create  => Just build them
                    install => Install them
                    ignore  => Ignore them


prereq_build    If true, always build the prereqs even if already
                resolved


verbose         Be verbose


force           Force the prereq to be built, even if already resolved
```