NAME

      Lintian::Profile − Profile parser for Lintian

SYNOPSIS

```
    my $profile = Lintian::Profile->new ('debian');
```

DESCRIPTION

      Lintian::Profile handles finding, parsing and implementation of Lintian Profiles as well as loading the relevant Lintian checks.

INSTANCE METHODS

      `$prof->`**known_aliases()**

          Returns a hash with old names that have new names.

      `$prof->`profile_list

          Returns a list ref of the (normalized) names of the profile and its parents. The last element of the list is the name of the profile itself, the second last is its parent and so on.

          Note: This list reference and its contents should not be modified.

      `$prof->`name

          Returns the name of the profile, which may differ from the name used to create this instance of the profile (e.g. due to symlinks).

      load ([$profname[, `$ipath`[, `$extra`]]])

          Loads a new profile. `$profname` is the name of the profile and `$ipath` is a list reference containing the path to one (or more) Lintian ''roots''.

          If `$profname` is undef, the default vendor will be loaded based on Dpkg::Vendor::get_current_vendor.

          If `$ipath` is not given, a default one will be used.

      `$prof->`known_tags
      `$prof->`enabled_tags
      `$prof->`get_taginfo ($name)

          Returns the Lintian::Tag::Info for `$tag` if known. Otherwise it returns undef.

      `$prof->`is_overridable ($tag)

          Returns a false value if the tag has been marked as ''non-overridable''. Otherwise it returns a truth value.

      `$prof->`known_checks
      `$prof->`enabled_checks
      `$prof->`get_checkinfo ($name)

          Returns the Lintian::Check::Info for `$name`. Otherwise it returns undef.

      `$prof->`enable_tag ($name)

          Enables a tag.

      `$prof->`disable_tag ($name)

          Disable a tag.

      `$prof->`include_path ([$path])

          Returns an array of paths to the (partial) Lintian roots, which are used by this profile. The paths are ordered from ''highest'' to ''lowest'' priority (i.e. items in the earlier paths should shadow those in later ones).

          If `$path` is given, the array will contain the paths to the path in these roots denoted by `$path`.

          Paths returned are not guaranteed to exists.

      display_level_for_tag

tag_is_enabled(TAG)

display(OPERATION, RELATION, SEVERITY)

> Configure which tags are displayed by severity. OPERATION is + to display the indicated tags, − to not display the indicated tags, or = to not display any tags except the indicated ones. RELATION is one of <, <=, =, >=, or >. The OPERATION will be applied to all values of severity that match the given RELATION on the SEVERITY argument. If either of those arguments are undefined, the action applies to any value for that variable. For example:

> ```
> $tags->display('=', '>=', 'error');
> ```

> turns off display of all tags and then enables display of any tag of severity error or higher.

> ```
> $tags->display('+', '>', 'warning');
> ```

> adds to the current configuration display of all tags with a severity higher than warning.

> ```
> $tags->display('-', '=', 'info');
> ```

> turns off display of tags of severity info.

> This method throws an exception on errors, such as an unknown severity or an impossible constraint (like > `serious`).

AUTHOR

> Originally written by Niels Thykier <niels@thykier.net> for Lintian.

SEE ALSO

> **lintian** (1)