

NAME

Dpkg::BuildFlags – query build flags

DESCRIPTION

This class is used by dpkg-buildflags and can be used to query the same information.

METHODS

`$bf = Dpkg::BuildFlags->new()`

Create a new Dpkg::BuildFlags object. It will be initialized based on the value of several configuration files and environment variables.

`$bf->load_vendor_defaults()`

Reset the flags stored to the default set provided by the vendor.

`$bf->load_system_config()`

Update flags from the system configuration.

`$bf->load_user_config()`

Update flags from the user configuration.

`$bf->load_environment_config()`

Update flags based on user directives stored in the environment. See **dpkg-buildflags**(1) for details.

`$bf->load_maintainer_config()`

Update flags based on maintainer directives stored in the environment. See **dpkg-buildflags**(1) for details.

`$bf->load_config()`

Call successively **load_system_config()**, **load_user_config()**, **load_environment_config()** and **load_maintainer_config()** to update the default build flags defined by the vendor.

`$bf->unset($flag)`

Unset the build flag `$flag`, so that it will not be known anymore.

`$bf->set($flag, $value, $source, $maint)`

Update the build flag `$flag` with value `$value` and record its origin as `$source` (if defined). Record it as maintainer modified if `$maint` is defined and true.

`$bf->set_feature($area, $feature, $enabled)`

Update the boolean state of whether a specific feature within a known feature area has been enabled. The only currently known feature areas are “future”, “qa”, “sanitize”, “hardening” and “reproducible”.

`$bf->strip($flag, $value, $source, $maint)`

Update the build flag `$flag` by stripping the flags listed in `$value` and record its origin as `$source` (if defined). Record it as maintainer modified if `$maint` is defined and true.

`$bf->append($flag, $value, $source, $maint)`

Append the options listed in `$value` to the current value of the flag `$flag`. Record its origin as `$source` (if defined). Record it as maintainer modified if `$maint` is defined and true.

`$bf->prepend($flag, $value, $source, $maint)`

Prepend the options listed in `$value` to the current value of the flag `$flag`. Record its origin as `$source` (if defined). Record it as maintainer modified if `$maint` is defined and true.

`$bf->update_from_conf($file, $source)`

Update the current build flags based on the configuration directives contained in `$file`. See **dpkg-buildflags**(1) for the format of the directives.

`$source` is the origin recorded for any build flag set or modified.

`$bf->get($flag)`

Return the value associated to the flag. It might be undef if the flag doesn't exist.

`$bf->get_feature_areas()`

Return the feature areas (i.e. the area values `has_features` will return true for).

`$bf->get_features($area)`

Return, for the given area, a hash with keys as feature names, and values as booleans indicating whether the feature is enabled or not.

`$bf->get_origin($flag)`

Return the origin associated to the flag. It might be undef if the flag doesn't exist.

`$bf->is_maintainer_modified($flag)`

Return true if the flag is modified by the maintainer.

`$bf->has_features($area)`

Returns true if the given area of features is known, and false otherwise. The only currently recognized feature areas are "future", "qa", "sanitize", "hardening" and "reproducible".

`$bf->has($option)`

Returns a boolean indicating whether the flag exists in the object.

`@flags = $bf->list()`

Returns the list of flags stored in the object.

CHANGES

Version 1.04 (dpkg 1.20.0)

New method: `$bf->unset()`.

Version 1.03 (dpkg 1.16.5)

New method: `$bf->get_feature_areas()` to list possible values for `$bf->get_features`.

New method `$bf->is_maintainer_modified()` and new optional parameter to `$bf->set()`, `$bf->append()`, `$bf->prepend()`, `$bf->strip()`.

Version 1.02 (dpkg 1.16.2)

New methods: `$bf->get_features()`, `$bf->has_features()`, `$bf->set_feature()`.

Version 1.01 (dpkg 1.16.1)

New method: `$bf->prepend()` very similar to `append()`. Implement support of the prepend operation everywhere.

New method: `$bf->load_maintainer_config()` that update the build flags based on the package maintainer directives.

Version 1.00 (dpkg 1.15.7)

Mark the module as public.