

**NAME**

ProjectBuilder::Distribution, part of the project-builder.org – module dealing with distribution detection

**DESCRIPTION**

This modules provides functions to allow detection of Linux distributions, and giving back some attributes concerning them.

**SYNOPSIS**

```
use ProjectBuilder::Distribution;

#
# Return information on the running distro
#
my $pbos = pb_distro_get_context();
print "distro tuple: ".Dumper($pbos->name, $pbos->ver, $pbos->fam, $pbos->type, $p
#
# Return information on the requested distro
#
my $pbos = pb_distro_get_context("ubuntu-7.10-x86_64");
print "distro tuple: ".Dumper($pbos->name, $pbos->ver, $pbos->fam, $pbos->type, $p
#
# Return information on the running distro
#
my ($ddir,$dver) = pb_distro_guess();
```

**USAGE****pb\_distro\_api**

This function returns the mandatory configuration file used for api

**pb\_distro\_conf**

This function returns the mandatory configuration file used for distribution/OS detection

**pb\_distro\_sysconf**

This function returns the optional configuration file used for local customization

**pb\_distro\_init**

This function returns a hash of parameters indicating the distribution name, version, family, type of build system, suffix of packages, update command line, installation command line and architecture of the underlying Linux distribution. The value of the fields may be “unknown” in case the function was unable to recognize on which distribution it is running.

As an example, Ubuntu and Debian are in the same “du” family. As well as RedHat, RHEL, CentOS, fedora are on the same “rh” family. Mandriva, Open SuSE and Fedora have all the same “rpm” type of build system. Ubuntu and Debian have the same “deb” type of build system. And “fc” is the extension generated for all Fedora packages (Version will be added by pb). All this information is stored in an external configuration file typically at /etc/pb/pb.yml

When passing the distribution name and version as parameters, the **pb\_distro\_init** function returns the parameter of that distribution instead of the underlying one.

Cf: <http://linuxmafia.com/faq/Admin/release-files.html> Ideas taken from <http://search.cpan.org/~kerberus/Linux-Distribution-0.14/lib/Linux/Distribution.pm>

**pb\_distro\_guess**

This function returns a list of 2 parameters indicating the distribution name and version of the underlying Linux distribution. The value of those 2 fields may be “unknown” in case the function was unable to recognize on which distribution it is running.

On my home machine it would currently report (“mandriva”, “2010.2”).

**pb\_distro\_getlsb**

This function returns the 5 lsb values LSB version, distribution ID, Description, release and codename. As entry it takes an optional parameter to specify whether the output is short or not.

**pb\_distro\_installpkgs**

This function install the packages passed as parameters on a distribution.

**pb\_distro\_installdeps**

This function install the dependencies required to build the package on a distro. If `$forcerepo` is defined then do not assume packages are already installed, but reinstall them (useful if you add a repo which contains more up to date packages that you need) Dependencies can be passed as the 4th parameter in which case they are not computed

**pb\_distro\_getdeps**

This function computes the dependencies indicated in the build file and return them as a string of packages to install

**pb\_distro\_only\_deps\_needed**

This function returns only the dependencies not yet installed

**pb\_distro\_setuposrepo**

This function sets up potential additional repository for the setup phase

**pb\_distro\_setuprepo**

This function sets up potential additional repository to the build/install/test environment If done, it returns forcerepo if a repo was added, if not undef

**pb\_distro\_setuprepo\_gen**

This functionthe sets up in a generic way potential additional repository passed as a param It returns forcerepo if one was added, else undef

**pb\_distro\_to\_keylist**

Given a pbos object (first param) and the generic key (second param), get the list of possible keys for looking up variable for filter names. The list will be sorted most-specific to least specific.

**pb\_distro\_get\_param**

This internal function gets the parameters in the conf file from the most precise tuple up to default

**pb\_distro\_get\_if**

This function gets the parameters in the conf file from the most precise tuple up to default

**pb\_distro\_get**

This function gets the parameters in the conf file from the most precise tuple up to default. Aborts of one param doesn't exist whereas it should

**pb\_distro\_get\_in\_hash\_if**

This function gets the parameters in the conf file passed as hash from the most precise tuple up to default

**pb\_distro\_get\_context**

This function gets the OS context passed as parameter and return the corresponding distribution hash If passed undef or "" then auto-detects

**pb\_distro\_conf\_print**

This function prints every configuration parameter in order to help debug stacking issues with conf files. If a VM/VE/RM is given, restrict display to this distribution. If parameters are passed, restrict again the display to these values only.

**WEB SITES**

The main Web site of the project is available at <<http://www.project-builder.org/>>. Bug reports should be filled using the trac instance of the project at <<http://trac.project-builder.org/>>.

**USER MAILING LIST**

None exists for the moment.

**AUTHORS**

The Project-Builder.org team <<http://trac.project-builder.org/>> lead by Bruno Cornec <<mailto:bruno@project-builder.org>>.

**COPYRIGHT**

Project-Builder.org is distributed under the GPL v2.0 license described in the file COPYING included with the distribution.