Name

command-not-found — A command-not-found handler

Synopsis

`command-not-found` {binary_name} {repository}

Arguments

The following arguments are required:

`binary_name`

The name of the binary you are looking for.

`repository`

The name of the repository to search. For most cases, use zypp

Description

the command-not-found handler searches the package a missing command belongs to.

The handler is integrated into the bash(1) and zsh(1) shells and doesn't need to be called directly. Just type the name of a command in your favourite shell and you'll get a result.

If you consider the c-n-f handler useless, just add `unset command_not_found_handle` to your profile or remove the command-not-found package.

The handler doesn't call the command-not-found binary directly, it only prints info about it. If you want to invoke it automatically, just add `export COMMAND_NOT_FOUND_AUTO=1` to your bash profile.

If you want to install found packages, add `export COMMAND_NOT_FOUND_INSTALL=1` to your bash profile.

Example: Normal usage

For example you want to try `blender`, because you have heard that is an amazing program. So just type blender in your shell:

```
$ blender
```

You get the following output:

```
The program 'blender' can be found in the following package:
  * blender [ path: /usr/bin/blender, repository: zypp (openSUSE 11.1-0) ]

Try installing with:
    sudo zypper install blender

bash: blender: command not found
```

See also

scout(1)