NAME

pg_config - retrieve information about the installed version of PostgreSQL

SYNOPSIS

pg_config [*option*...]

DESCRIPTION

The pg_config utility prints configuration parameters of the currently installed version of PostgreSQL. It is intended, for example, to be used by software packages that want to interface to PostgreSQL to facilitate finding the required header files and libraries.

OPTIONS

To use pg_config, supply one or more of the following options:

--bindir

Print the location of user executables. Use this, for example, to find the **psql** program. This is normally also the location where the pg_config program resides.

--docdir

Print the location of documentation files.

--htmldir

Print the location of HTML documentation files.

--includedir

Print the location of C header files of the client interfaces.

--pkgincludedir

Print the location of other C header files.

--includedir-server

Print the location of C header files for server programming.

--libdir

Print the location of object code libraries.

--pkglibdir

Print the location of dynamically loadable modules, or where the server would search for them. (Other architecture-dependent data files might also be installed in this directory.)

--localedir
Print the location of locale support files. (This will be an empty string if locale support was not configured when PostgreSQL was built.)

--mandir
Print the location of manual pages.

--sharedir
Print the location of architecture-independent support files.

--sysconfdir
Print the location of system-wide configuration files.

--pgxs
Print the location of extension makefiles.

--configure
Print the options that were given to the configure script when PostgreSQL was configured for building. This can be used to reproduce the identical configuration, or to find out with what options a binary package was built. (Note however that binary packages often contain vendor-specific custom patches.) See also the examples below.

--cc
Print the value of the *CC* variable that was used for building PostgreSQL. This shows the C compiler used.

--cppflags
Print the value of the *CPPFLAGS* variable that was used for building PostgreSQL. This shows C compiler switches needed at preprocessing time (typically, -I switches).

--cflags
Print the value of the *CFLAGS* variable that was used for building PostgreSQL. This shows C compiler switches.

--cflags_sl
Print the value of the *CFLAGS_SL* variable that was used for building PostgreSQL. This shows extra C compiler switches used for building shared libraries.

--ldflags
Print the value of the *LDFLAGS* variable that was used for building PostgreSQL. This shows

linker switches.

--ldflags_ex
Print the value of the *LDFLAGS_EX* variable that was used for building PostgreSQL. This shows linker switches used for building executables only.

--ldflags_sl
Print the value of the *LDFLAGS_SL* variable that was used for building PostgreSQL. This shows linker switches used for building shared libraries only.

--libs
Print the value of the *LIBS* variable that was used for building PostgreSQL. This normally contains -l switches for external libraries linked into PostgreSQL.

--version
Print the version of PostgreSQL.

-?
--help
Show help about pg_config command line arguments, and exit.

If more than one option is given, the information is printed in that order, one item per line. If no options are given, all available information is printed, with labels.

NOTES

The options **--docdir**, **--pkgincludedir**, **--localedir**, **--mandir**, **--sharedir**, **--sysconfdir**, **--cc**, **--cppflags**, **--cflags**, **--cflags_sl**, **--ldflags**, **--ldflags_sl**, and **--libs** were added in PostgreSQL 8.1. The option **--htmldir** was added in PostgreSQL 8.4. The option **--ldflags_ex** was added in PostgreSQL 9.0.

EXAMPLE

To reproduce the build configuration of the current PostgreSQL installation, run the following command:

```
eval ./configure ‘pg_config --configure‘
```

The output of pg_config --configure contains shell quotation marks so arguments with spaces are represented correctly. Therefore, using eval is required for proper results.