NAME

oid2name - resolve OIDs and file nodes in a PostgreSQL data directory

SYNOPSIS

oid2name [*option*...]

DESCRIPTION

oid2name is a utility program that helps administrators to examine the file structure used by PostgreSQL. To make use of it, you need to be familiar with the database file structure, which is described in Chapter 73.

> **Note**
>
> The name "oid2name" is historical, and is actually rather misleading, since most of the time when you use it, you will really be concerned with tables' filenode numbers (which are the file names visible in the database directories). Be sure you understand the difference between table OIDs and table filenodes!

oid2name connects to a target database and extracts OID, filenode, and/or table name information. You can also have it show database OIDs or tablespace OIDs.

OPTIONS

oid2name accepts the following command-line arguments:

-f *filenode*
--filenode=*filenode*
show info for table with filenode *filenode*.

-i
--indexes
include indexes and sequences in the listing.

-o *oid*
--oid=*oid*
show info for table with OID *oid*.

-q
--quiet
omit headers (useful for scripting).

-s
--tablespaces
show tablespace OIDs.

-S
--system-objects
include system objects (those in **information_schema**, **pg_toast** and **pg_catalog** schemas).

-t *tablename_pattern*
--table=*tablename_pattern*
show info for table(s) matching *tablename_pattern*.

-V
--version
Print the oid2name version and exit.

-x
--extended
display more information about each object shown: tablespace name, schema name, and OID.

-?
--help
Show help about oid2name command line arguments, and exit.

oid2name also accepts the following command-line arguments for connection parameters:

-d *database*
--dbname=*database*
database to connect to.

-h *host*
--host=*host*
database server's host.

-H *host*
database server's host. Use of this parameter is *deprecated* as of PostgreSQL 12.

-p *port*
--port=*port*
database server's port.

-U *username*
--username=*username*
user name to connect as.

To display specific tables, select which tables to show by using **-o**, **-f** and/or **-t**. **-o** takes an OID, **-f** takes a filenode, and **-t** takes a table name (actually, it's a LIKE pattern, so you can use things like foo%). You can use as many of these options as you like, and the listing will include all objects matched by any of the options. But note that these options can only show objects in the database given by **-d**.

If you don't give any of **-o**, **-f** or **-t**, but do give **-d**, it will list all tables in the database named by **-d**. In this mode, the **-S** and **-i** options control what gets listed.

If you don't give **-d** either, it will show a listing of database OIDs. Alternatively you can give **-s** to get a tablespace listing.

ENVIRONMENT

PGHOST
PGPORT
PGUSER
Default connection parameters.

This utility, like most other PostgreSQL utilities, also uses the environment variables supported by libpq (see Section 34.15).

The environment variable **PG_COLOR** specifies whether to use color in diagnostic messages. Possible values are always, auto and never.

NOTES

oid2name requires a running database server with non-corrupt system catalogs. It is therefore of only limited use for recovering from catastrophic database corruption situations.

EXAMPLES

```
$ # what's in this database server, anyway?
$ oid2name
All databases:
    Oid  Database Name  Tablespace
----------------------------------
  17228       alvherre  pg_default
  17255     regression  pg_default
```

```
  17227      template0  pg_default
      1      template1  pg_default

$ oid2name -s
All tablespaces:
     Oid  Tablespace Name
------------------------
    1663       pg_default
    1664        pg_global
  155151         fastdisk
  155152          bigdisk

$ # OK, let's look into database alvherre
$ cd $PGDATA/base/17228

$ # get top 10 db objects in the default tablespace, ordered by size
$ ls -lS * | head -10
-rw-------  1 alvherre alvherre 136536064 sep 14 09:51 155173
-rw-------  1 alvherre alvherre  17965056 sep 14 09:51 1155291
-rw-------  1 alvherre alvherre   1204224 sep 14 09:51 16717
-rw-------  1 alvherre alvherre    581632 sep  6 17:51 1255
-rw-------  1 alvherre alvherre    237568 sep 14 09:50 16674
-rw-------  1 alvherre alvherre    212992 sep 14 09:51 1249
-rw-------  1 alvherre alvherre    204800 sep 14 09:51 16684
-rw-------  1 alvherre alvherre    196608 sep 14 09:50 16700
-rw-------  1 alvherre alvherre    163840 sep 14 09:50 16699
-rw-------  1 alvherre alvherre    122880 sep  6 17:51 16751

$ # I wonder what file 155173 is ...
$ oid2name -d alvherre -f 155173
From database "alvherre":
  Filenode  Table Name
----------------------
    155173    accounts

$ # you can ask for more than one object
$ oid2name -d alvherre -f 155173 -f 1155291
From database "alvherre":
  Filenode     Table Name
------------------------
```

```
  155173       accounts
 1155291  accounts_pkey

$ # you can mix the options, and get more details with -x
$ oid2name -d alvherre -t accounts -f 1155291 -x
From database "alvherre":
  Filenode     Table Name      Oid  Schema  Tablespace
----------------------------------------------------
  155173       accounts   155173  public  pg_default
 1155291  accounts_pkey  1155291  public  pg_default

$ # show disk space for every db object
$ du [0-9]* |
> while read SIZE FILENODE
> do
>   echo "$SIZE       ‘oid2name -q -d alvherre -i -f $FILENODE‘"
> done
16          1155287  branches_pkey
16          1155289  tellers_pkey
17561          1155291  accounts_pkey
...

$ # same, but sort by size
$ du [0-9]* | sort -rn | while read SIZE FN
> do
>   echo "$SIZE   ‘oid2name -q -d alvherre -f $FN‘"
> done
133466           155173    accounts
17561          1155291  accounts_pkey
1177           16717  pg_proc_proname_args_nsp_index
...

$ # If you want to see what’s in tablespaces, use the pg_tblspc directory
$ cd $PGDATA/pg_tblspc
$ oid2name -s
All tablespaces:
    Oid  Tablespace Name
-----------------------
   1663       pg_default
   1664        pg_global
```

```
  155151         fastdisk
  155152          bigdisk

$ # what databases have objects in tablespace "fastdisk"?
$ ls -d 155151/*
155151/17228/  155151/PG_VERSION

$ # Oh, what was database 17228 again?
$ oid2name
All databases:
    Oid  Database Name  Tablespace
----------------------------------
  17228       alvherre  pg_default
  17255     regression  pg_default
  17227      template0  pg_default
      1      template1  pg_default

$ # Let's see what objects does this database have in the tablespace.
$ cd 155151/17228
$ ls -l
total 0
-rw-------  1 postgres postgres 0 sep 13 23:20 155156

$ # OK, this is a pretty small table ... but which one is it?
$ oid2name -d alvherre -f 155156
From database "alvherre":
  Filenode  Table Name
----------------------
    155156         foo
```

AUTHOR

B. Palmer <bpalmer@crimelabs.net>