
Postgres Toolkit Documentation

Release 0.3dev

Uptime Technologies, LLC

May 15, 2018

1	Install the Toolkit	3
1.1	Supported OS	3
1.2	PostgreSQL Version	3
1.3	Installation	3
2	pt-archive-xlog	5
2.1	Summary	5
2.2	Usage	5
2.3	Options	5
2.4	Output Items	5
2.5	Examples	5
3	pt-config	7
3.1	Summary	7
3.2	Usage	7
3.3	Commands	7
3.4	Options	7
3.5	Output Items	8
3.6	Examples	8
4	pt-index-usage	11
4.1	Summary	11
4.2	Usage	11
4.3	Options	11
4.4	Output Items	12
4.5	Examples	12
5	pt-kill	15
5.1	Summary	15
5.2	Usage	15
5.3	Commands	15
5.4	Options	15
5.5	Output Items	15
5.6	Examples	16
6	pt-privilege-autogen	17
6.1	Summary	17

6.2	Usage	17
6.3	Commands	17
6.4	Options	17
6.5	Output Items	18
6.6	Examples	18
7	pt-proc-stat	21
7.1	Summary	21
7.2	Usage	21
7.3	Options	21
7.4	Output Items	22
7.5	Examples	22
8	pt-replication-stat	23
8.1	Summary	23
8.2	Usage	23
8.3	Options	23
8.4	Output Items	24
8.5	Examples	24
9	pt-session-profiler	27
9.1	Summary	27
9.2	Usage	27
9.3	Options	27
9.4	Output Items	28
9.5	Examples	28
10	pt-set-tablespace	29
10.1	Summary	29
10.2	Usage	29
10.3	Options	29
10.4	Output Items	30
10.5	Examples	30
11	pt-show-locks	33
11.1	Summary	33
11.2	Usage	33
11.3	Options	33
11.4	Output Items	34
11.5	Examples	34
12	pt-snap-statements	35
12.1	Summary	35
12.2	Usage	35
12.3	Options	35
12.4	Output Items	36
12.5	Examples	36
13	pt-stat-snapshot	39
13.1	Summary	39
13.2	Usage	39
13.3	Commands	39
13.4	Options	40
13.5	Output Items	40
13.6	Examples	40

13.7	About pgperf-snapshot Module	41
14	pt-table-usage	57
14.1	Summary	57
14.2	Usage	57
14.3	Options	57
14.4	Output Items	58
14.5	Examples	58
15	pt-tablespace-usage	61
15.1	Summary	61
15.2	Usage	61
15.3	Options	61
15.4	Output Items	62
15.5	Examples	62
16	pt-verify-checksum	63
16.1	Summary	63
16.2	Usage	63
16.3	Options	64
16.4	Output Items	64
16.5	Examples	64
17	pt-xact-stat	65
17.1	Summary	65
17.2	Usage	65
17.3	Options	65
17.4	Output Items	66
17.5	Examples	66
18	Release Note	69
18.1	Version 0.2.1	69
18.2	Version 0.2.0	69
19	Indices and tables	71

Postgres Toolkit is a collection of script and command that can easily perform on the complex DBA works when performing PostgreSQL server operation management, performance tuning and Troubleshooting.

If you have any questions, requests or report bugs about Postgres Toolkit, please email me or use Github issues. email:postgres-toolkit at uptime dot jp Github: <http://www.github.com/uptimejp/postgres-toolkit/>

Contents:

1.1 Supported OS

For a list of Operating System supported.

- Red Hat Enterprise Linux 6 / CentOS 6
- Red Hat Enterprise Linux 7 / CentOS 7
- Ubuntu 14.04 LTS

Make sure you have Python2.6 or Python2.7 is installed.

1.2 PostgreSQL Version

For a list of PostgreSQL supported.

- PostgreSQL 9.0
- PostgreSQL 9.1
- PostgreSQL 9.2
- PostgreSQL 9.3
- PostgreSQL 9.4

1.3 Installation

Use following command, you can install the toolkit via the Internet.

Run

```
curl -L http://dl.uptimeforce.com/postgres-toolkit/install.sh | sh
```

or

```
wget http://dl.uptimeforce.com/postgres-toolkit/install.sh  
sh install.sh
```

When the installs are done, copying related files under `/opt/uptime/postgres-toolkit-<VERSION>`.

2.1 Summary

Archive a transaction log file in safe way.

This command can be used in the `archive_command` parameter.

2.2 Usage

```
pt-archive-xlog <XLOGFILEPATH> <DESTDIR>
```

2.3 Options

No option.

2.4 Output Items

It returns with the exit code 0 on success, or 1 on failure.

2.5 Examples

Following example shows how to configure `archive_command` in `postgresql.conf`.

```
archive_command = '/path/to/pt-archive-xlog %p /path/to/archivedir'
```


3.1 Summary

Refer the setting of the PostgreSQL configuration file `postgresql.conf`, and then modify it.

3.2 Usage

```
pt-config [options...] get [PARAM]
pt-config [options...] set [PARAM] [VALUE]
pt-config [options...] disable [PARAM]
```

3.3 Commands

<code>get [PARAM]</code>	Displays the current value. If it is disabled(comment-out). (disabled) is displayed.
<code>set [PARAM] [VALUE]</code>	Sets the new value. The value will be enabled if it is disabled (comment-out).
<code>disable [PARAM]</code>	Disable the setting value.(comment-out)

3.4 Options

```
-D, --pgdata=PGDATA    Specify a PostgreSQL database cluster.
--apply                 Apply change(s).
--help                  Print this help.
```

-D, --pgdata Specifies the PostgreSQL database cluster. If not specified, to set the value of PGDATA.
 --apply It will actually apply the changes to the postgresql.conf when it runs set / disable command.

3.5 Output Items

Reading:	Loading the postgresql.conf file displays in the full path.
Dry-run mode:	The before and after values is only displayed. It does not provide the actual configuration changes.
Applying:	Doing the actual configuration changes.
Old	Displays the value of before change.
New	Displays the value of after change.
Updating:	Modifying the postgresql.conf file displays in the full path.

3.6 Examples

Displays the current value of shared_buffers

```
$ pt-config -D /var/lib/pgsql/9.4/data get shared_buffers
[2015-04-16 17:08:12] INFO: Reading: /var/lib/pgsql/9.4/data/postgresql.conf
256MB
$
```

Sets the value 512MB of shared_buffers. (Does not do the actual configuration changes)

```
$ pt-config -D /var/lib/pgsql/9.4/data set shared_buffers 512MB
[2015-04-16 17:08:44] INFO: Reading: /var/lib/pgsql/9.4/data/postgresql.conf
[2015-04-16 17:08:44] INFO: Dry-run mode:
[2015-04-16 17:08:44] INFO:   Old: shared_buffers = 256MB                # min_
↔128kB
[2015-04-16 17:08:44] INFO:   New: shared_buffers = 512MB              # min_
↔128kB
$
```

Sets the value 512MB of shared_buffers. (Doing the actual configuration change)

```
$ pt-config -D /var/lib/pgsql/9.4/data --apply set shared_buffers 512MB
[2015-04-16 17:09:11] INFO: Reading: /var/lib/pgsql/9.4/data/postgresql.conf
[2015-04-16 17:09:11] INFO: Applying:
[2015-04-16 17:09:11] INFO:   Old: shared_buffers = 256MB                # min_
↔128kB
[2015-04-16 17:09:11] INFO:   New: shared_buffers = 512MB              # min_
↔128kB
[2015-04-16 17:09:11] INFO: Updated: /var/lib/pgsql/9.4/data/postgresql.conf
$
```

Sets the disable(comment-out) of shared_buffers.

```
$ pt-config -D /var/lib/pgsql/9.4/data --apply disable shared_buffers
[2015-04-16 17:09:52] INFO: Reading: /var/lib/pgsql/9.4/data/postgresql.conf
[2015-04-16 17:09:52] INFO: Applying:
[2015-04-16 17:09:52] INFO:   Old: shared_buffers = 512MB                # min_
↔128kB
```

(continues on next page)

(continued from previous page)

```
[2015-04-16 17:09:52] INFO:   New: #shared_buffers = 512MB           # min_
↪128kB
[2015-04-16 17:09:52] INFO: Updated: /var/lib/pgsql/9.4/data/postgresql.conf
$
```

Displays the value of `shared_buffers` (Disabled).

```
$ pt-config -D /var/lib/pgsql/9.4/data get shared_buffers
[2015-04-16 17:10:00] INFO: Reading: /var/lib/pgsql/9.4/data/postgresql.conf
512MB (disabled)
$
```


4.1 Summary

Displays the indexes usage.

4.2 Usage

```
pt-index-usage [option...]
```

4.3 Options

```
-h, --host=HOSTNAME  
-p, --port=PORT  
-U, --username=USERNAME  
-d, --dbname=DBNAME  
-o, --owner=STRING  
-n, --schema=STRING  
-t, --table=STRING  
-i, --index=STRING  
-u, --unused  
--help
```

-h, --host Specifies the connecting the PostgreSQL database server name or its IP address. If not specified, to set the value of PGHOST. localhost will be used if no other value has been defined in PGHOST.

-p, --port Specifies the port number of the connecting PostgreSQL database. If not specified, to set the value of PGPORT. 5432 will be used if no value has been defined in PGPORT.

-U, --username Specifies the user name of the PostgreSQL database. If not specified, to set the value of PGUSER. The value of USER will be used If no other value has been defined in PGUSER.

- d, --dbname Specifies the connecting database name. If not specified, to set the value of PGDATABASE. Database name as database user name will be used if no other value has been defined in PGDATABASE.
 - o, --owner The index information is only displayed which matches to a string specified where is a name of the owner. Also accepts regular expression when starting and ending with slashes. (cf. /REGEXP/)
 - n, --schema The index information is only displayed which matches to a string specified where is in the schema. Also accepts regular expression when starting and ending with slashes. (cf. /REGEXP/)
 - t, --table The index information is only displayed which matches to a string specified where is in the table. Also accepts regular expression when starting and ending with slashes. (cf. /REGEXP/)
 - i, --index The index information is only displayed which matches to a string specified. Also accepts regular expression when starting and ending with slashes. (cf. /REGEXP/)
 - u, --unused The index information is displayed which not use the index.
- d (or --dbname), -o (or --owner), -n (or --schema), -t (or --table), -i (or --index), -u (or --unused), if specified the options at the same time, the index is displayed which matches to it.

4.4 Output Items

OID	Object ID of index
OWNER	Owner name of index
SCHEMA	Schema name on being indexed
TABLE	Table name on being indexed
INDEX	Index name
BLKS	Block (every 8kb) of index
SCAN	Index scan number of executions
T_READ	Number of index entries by index scan
T_FTCH	Number of tuple on table by index scan
B_READ	Number of index blocks read from disk
B_HIT	Number of index pages read from shared buffer
STATUS	Index status. It reads from pg_index system table.
TABLESPACE	The tablespace name that has index

4.5 Examples

For the index of table in the public schema, displays the usage situation.

```
$ pt-index-usage -n public -d postgres
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|  OID  | OWNER | SCHEMA | TABLE | INDEX | BLKS | SCAN | T_ |
| READ | T_FTCH | B_READ | B_HIT | STATUS | TABLESPACE |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 26793 | snaga | public | pgbench_accounts | pgbench_accounts_pkey | 276 | 1 | 1 |
| 1 | 1 | 4 | 0 | | spc1 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 26789 | snaga | public | pgbench_branches | pgbench_branches_pkey | 2 | 1 | 1 |
| 1 | 0 | 2 | 0 | | pg_default |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 26791 | snaga | public | pgbench_tellers | pgbench_tellers_pkey | 2 | 0 | 1 |
| 0 | 0 | 0 | 0 | | pg_default |
```

(continues on next page)

(continued from previous page)

```

+-----+-----+-----+-----+-----+-----+-----+-----+
↪-----+-----+-----+-----+-----+-----+
$

```

For the index of `pgbench_accounts` table in the `public` schema, displays the usage situation.

```

$ pt-index-usage -n public -d postgres -t pgbench_accounts
+-----+-----+-----+-----+-----+-----+-----+-----+
↪-----+-----+-----+-----+-----+-----+
|  OID  | OWNER | SCHEMA |      TABLE      |      INDEX      | BLKS | SCAN | T_
↪READ | T_FTCH | B_READ | B_HIT | STATUS | TABLESPACE |
+-----+-----+-----+-----+-----+-----+-----+-----+
↪-----+-----+-----+-----+-----+-----+
| 26793 | snaga | public | pgbench_accounts | pgbench_accounts_pkey | 276 | 1 |
↪ 1 | 1 | 4 | 0 | | spc1 |
+-----+-----+-----+-----+-----+-----+-----+-----+
↪-----+-----+-----+-----+-----+-----+
$

```

For the index not used even once, in the `public` schema, displays the usage situation.

```

$ pt-index-usage -d postgres -n public -u
+-----+-----+-----+-----+-----+-----+-----+-----+
↪-----+-----+-----+-----+-----+-----+
|  OID  | OWNER | SCHEMA |      TABLE      |      INDEX      | BLKS | SCAN | T_
↪READ | T_FTCH | B_READ | B_HIT | STATUS | TABLESPACE |
+-----+-----+-----+-----+-----+-----+-----+-----+
↪-----+-----+-----+-----+-----+-----+
| 26791 | snaga | public | pgbench_tellers | pgbench_tellers_pkey | 2 | 0 |
↪ 0 | 0 | 0 | 0 | | pg_default |
+-----+-----+-----+-----+-----+-----+-----+-----+
↪-----+-----+-----+-----+-----+-----+
$

```


5.1 Summary

Interrupting the processing of PostgreSQL backend, or to exit it.

5.2 Usage

```
pt-kill [options...] [command] [pid]
```

5.3 Commands

cancel	Cancel a running query.
terminate	Terminate a backend with canceling query.

5.4 Options

--help	Print this help.
--------	------------------

5.5 Output Items

None.

5.6 Examples

Cancel the SQL running in the process ID 3289.

```
$ pt-kill cancel 3289
```

Exit the backend in the process ID 3219.

```
$ pt-kill terminate 3291
```

pt-privilege-autogen

6.1 Summary

Generates REVOKE/GRANT statements for giving minimum privileges to users and database objects. It uses PostgreSQL statistics to learn access policy of the application.

6.2 Usage

```
pt-privilege-autogen [option...] [ start | stop ]
pt-privilege-autogen [option...] generate <USERNAME>
```

6.3 Commands

start	Starts collecting table access statistics of the database.
generate	Generates REVOKE/GRANT statements from collected access statistics of the tables. Requires user name to be granted.
stop	Stops collecting table access statistics of the database.

6.4 Options

```
-h, --host=HOSTNAME
-p, --port=PORT
-U, --username=USERNAME
-d, --dbname=DBNAME
--help
```

`-h, --host` Specifies the connecting PostgreSQL database server name or IP address. If not specified, to set the value of `PGHOST`. `localhost` will be used if no value has been defined in `PGHOST`.

`-p, --port` Specifies the port number of the connecting PostgreSQL database. If not specified, to set the value of `PGPORT`. `5432` will be used if no value has been defined in `PGPORT`.

`-U, --username` Specifies the user name of the PostgreSQL database. If not specified, to set the value of `PGUSER`. The value of `USER` will be used if no value has been defined in `PGUSER`.

`-d, --dbname` Specifies the connecting database name. If not specified, to set the value of `PGDATABASE`. It connect to the database same as the database username if no value has been defined in `PGDATABASE`.

6.5 Output Items

This command shows `GRANT/REVOKE` statements that can be applied to the PostgreSQL database.

6.6 Examples

This example shows how to generate `REVOKE/GRANT` statements for the user `snaga` to be allowed to run `pgbench` transactions with minimum privileges.

Following procedure needs to be done by the user who already has the several permissions, like `super-user`. This example uses `postgres super-user`.

First, prepare a database for `pgbench` with `superuser`, and make sure that a regular user can't access to the tables.

```
$ createdb -U postgres mydb
$ pgbench -i -U postgres mydb
NOTICE: table "pgbench_history" does not exist, skipping
NOTICE: table "pgbench_tellers" does not exist, skipping
NOTICE: table "pgbench_accounts" does not exist, skipping
NOTICE: table "pgbench_branches" does not exist, skipping
creating tables...
100000 of 100000 tuples (100%) done (elapsed 0.70 s, remaining 0.00 s).
vacuum...
set primary keys...
done.
$ pgbench -c 1 -t 1 -U snaga -n mydb
ERROR: permission denied for relation pgbench_branches
$
```

Second, start collecting table access statistics, and run `pgbench` transactions.

```
$ pt-privilege-autogen -U postgres -d mydb start
[2015-08-04 04:40:45] INFO: Collecting access statistics started.
$ pgbench -c 1 -t 1 -U postgres -n mydb
transaction type: TPC-B (sort of)
scaling factor: 1
query mode: simple
number of clients: 1
number of threads: 1
number of transactions per client: 1
number of transactions actually processed: 1/1
latency average: 0.000 ms
tps = 14.402581 (including connections establishing)
```

(continues on next page)

(continued from previous page)

```
tps = 20.464964 (excluding connections establishing)
$
```

Then, generate an access policy file, and apply it to the database. A regular user `snaga` is granted least privileges for four tables here.

```
$ pt-privilege-autogen -U postgres -d mydb generate snaga

-- Database
REVOKE ALL ON DATABASE "mydb" FROM "public";
GRANT CONNECT,TEMP ON DATABASE "mydb" TO "snaga";

-- Schema
REVOKE ALL ON SCHEMA "public" FROM "public";
GRANT USAGE ON SCHEMA "public" TO "snaga";

-- Table
REVOKE ALL ON TABLE "public"."pgbench_accounts" FROM "snaga";
REVOKE ALL ON TABLE "public"."pgbench_branches" FROM "snaga";
REVOKE ALL ON TABLE "public"."pgbench_history" FROM "snaga";
REVOKE ALL ON TABLE "public"."pgbench_tellers" FROM "snaga";
GRANT SELECT,UPDATE ON TABLE "public"."pgbench_accounts" TO "snaga";
GRANT SELECT,UPDATE ON TABLE "public"."pgbench_branches" TO "snaga";
GRANT INSERT ON TABLE "public"."pgbench_history" TO "snaga";
GRANT SELECT,UPDATE ON TABLE "public"."pgbench_tellers" TO "snaga";

$ pt-privilege-autogen -U postgres -d mydb generate snaga > grant.sql
$ psql -f grant.sql -U postgres mydb
REVOKE
GRANT
REVOKE
GRANT
REVOKE
REVOKE
REVOKE
REVOKE
GRANT
GRANT
GRANT
GRANT
$
```

Finally, stop collecting access statistics, and make sure that the regular user `snaga` can now run `pgbench` transaction on the database with the least privileges.

```
$ pt-privilege-autogen -U postgres -d mydb stop
[2015-08-04 04:44:21] INFO: Collecting access statistics stopped.
$ pgbench -c 1 -t 1 -U snaga -n mydb
transaction type: TPC-B (sort of)
scaling factor: 1
query mode: simple
number of clients: 1
number of threads: 1
number of transactions per client: 1
number of transactions actually processed: 1/1
```

(continues on next page)

(continued from previous page)

```
latency average: 0.000 ms
tps = 33.598764 (including connections establishing)
tps = 82.182774 (excluding connections establishing)
$
```

7.1 Summary

Displays statistics I/O for each process.

7.2 Usage

```
pt-proc-stat [option...] [delay [count]]
```

`pt-proc-stat` Referring to proc file system. Make sure you run by running PostgreSQL user or root user.

7.3 Options

```
-D, --pgdata=DATADIR  
-P, --pid=PID  
--help
```

`-D, --pgdata` Specifies the directory of the database cluster.

`-P, --pid` Specifies the process ID of the postmaster process.

7.4 Output Items

PROCESS NAME	Process name
PID	Process ID
STAT	Status of the process
USR	User CPU time (Difference)
SYS	System CPU time (Difference)
VSZ	Virtual memory size (MB)
RSS	Usage of a physical memory (MB)
READ	Reading I/O volume is displayed differently between both of the data(kB).
WRITE	Writing I/O volume is displayed differently between both of the data(kB).
READ2	Load I/O except the READ (KB / Difference)
WRITE2	Load I/O except the WRITE (KB / Difference)

7.5 Examples

It looks for the postmaster process automatically. Statistics I/O of the postmaster and the child process is displayed every five seconds twice.

```
$ sudo ./pt-proc-stat 5 2
Fri May 1 22:23:39 JST 2015
  PROCESS NAME[ PID] STAT  USR  SYS  VSZ  RSS   READ  WRITE  READ2 WRITE2
    postmaster[24026]   S    4   13  100   9  23752 1290092 1090800 155357
      logger[24028]    S    0    1   85    1    4     76     30    -45
    checkpointer[24030] S    4  117  100   8   176  56768  -176 -46965
      writer[24031]    S  104   66  100   9    0  291080    0 130560
    wal writer[24032]  S    8   19  100   1    0   2928    0     0
  autovacuum launcher[24033] S    3    3  101   2    8     8   288    0
    stats collector[24034] S   13   32   85   1    0   2140    34  -649
snaga postgres 127.0[25473] R   32    9  101   7   296  1472  1264    0
snaga postgres 127.0[25474] R   33    9  101   7   424  1384  1120    0
snaga postgres 127.0[25475] R   33    9  101   7   424  1448  1016    0
snaga postgres 127.0[25476] S   32    9  101   7   580  1400   780    0
snaga postgres 127.0[25477] R   32    9  101   7   908  1368   492    0

Fri May 1 22:23:44 JST 2015
  PROCESS NAME[ PID] STAT  USR  SYS  VSZ  RSS   READ  WRITE  READ2 WRITE2
    postmaster[24026]   S    0    0  100   9    0     0     0     0
      logger[24028]    S    0    0   85    1    0     0     0     0
    checkpointer[24030] S    0    0  100   8    0     0     0     0
      writer[24031]    S    4    1  100   9    0  11928    0   392
    wal writer[24032]  S    0    0  100   1    0     0     0     0
  autovacuum launcher[24033] S    0    0  101   2    0     0     0     0
    stats collector[24034] S    0    0   85    1    0     0     0     0
snaga postgres 127.0[25473] R   72   18  101  10  1772  3608  1740    0
snaga postgres 127.0[25474] R   68   20  101  10  1436  3920  2020    0
snaga postgres 127.0[25475] D   70   18  101  10  1304  4216  2368    0
snaga postgres 127.0[25476] R   70   20  101  10  1252  3384  2212    0
snaga postgres 127.0[25477] R   73   16  101  10  1464  3224  2080    0

$
```

8.1 Summary

The replication status is displayed by referring the master node statistical information of the replication. It is possible to display continuously on every specified interval.

PostgreSQL 9.1 or higher. PostgreSQL 9.0 doesn't work because it doesn't have no "pg_stat_replication" system view.

8.2 Usage

```
pt-replication-stat [option...] [delay [count]]
```

8.3 Options

```
-h, --host=HOSTNAME  
-p, --port=PORT  
-U, --username=USERNAME  
-d, --dbname=DBNAME
```

8.4 Output Items

PID	The process ID of WAL sender processes on the master node
NAME	The slave node name of the registered in the replication
HOST	The host name of slave node or IP address
PORT	The port number of the master node that connected to the slave node
STATE	The slave node status. It acquires the value of following startup or backup or catchup or streaming
SENT	Position on the WAL which sent to slave
WRITTEN	Position on the WAL which wrote to the WAL buffer on the slave
FLUSHED	Position on the WAL which synchronously wrote to the WAL file on the slave
REPLAYED	Position on WAL which applied to the data file on the slave
PRI	If the slave node is synchronously replication that the node priority is displayed.
MODE	The operation mode is displayed. sync is synchronous mode. async is asynchronous mode. potential is running in asynchronous mode. But there is a possibility to upgrade to the synchronous mode.

8.5 Examples

Connect to the port 5433 of host 127.0.0.1 by postgres user, the statistical information is displayed every five seconds twice.

```
$ pt-replication-stat -h 127.0.0.1 -p 5433 -U postgres 5 2
Sat Mar 28 21:45:23 JST 2015
+-----+-----+-----+-----+-----+-----+-----+-----+
↪+-----+-----+
| PID | NAME | HOST | PORT | STATE | SENT | WRITTEN | FLUSHED |
↪| REPLAYED | PRI | MODE |
+-----+-----+-----+-----+-----+-----+-----+-----+
↪+-----+-----+
| | | | | local | 0/5F30398 | 0/5F300B0 | |
↪| | | | master |
| 3323 | replica1 | 127.0.0.1 | 55580 | streaming | 0/5F300B0 | 0/5F300B0 | 0/5F300B0 |
↪| 0/5F2FE48 | 0 | async |
| 3367 | replica2 | 127.0.0.1 | 55589 | streaming | 0/5F300B0 | 0/5F300B0 | 0/5F2FE48 |
↪| 0/5F2FE48 | 0 | async |
+-----+-----+-----+-----+-----+-----+-----+-----+
↪+-----+-----+

Sat Mar 28 21:45:28 JST 2015
+-----+-----+-----+-----+-----+-----+-----+-----+
↪+-----+-----+
| PID | NAME | HOST | PORT | STATE | SENT | WRITTEN | FLUSHED |
↪| REPLAYED | PRI | MODE |
+-----+-----+-----+-----+-----+-----+-----+-----+
↪+-----+-----+
| | | | | local | 0/608CD68 | 0/608CAC0 | |
↪| | | | master |
| 3323 | replica1 | 127.0.0.1 | 55580 | streaming | 0/608CAC0 | 0/608CAC0 | 0/608C7D8 |
↪| 0/608C7D8 | 0 | async |
| 3367 | replica2 | 127.0.0.1 | 55589 | streaming | 0/608CAC0 | 0/608CAC0 | 0/608C7D8 |
↪| 0/608C7D8 | 0 | async |
```

(continues on next page)

(continued from previous page)

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+
↪+-----+-----+
$

```

Connect to the default port (5432) of localhost, the statistical information is displayed every five seconds continuously. CTRL-C to end this.

```

$ pt-replication-stat -h localhost 5
Sat Mar 28 21:45:23 JST 2015
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
↪+-----+-----+
| PID | NAME | HOST | PORT | STATE | SENT | WRITTEN | FLUSHED |
↪| REPLAYED | PRI | MODE |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
↪+-----+-----+
| | | | | local | 0/5F30398 | 0/5F300B0 | |
↪| | | | master |
| 3323 | replica1 | 127.0.0.1 | 55580 | streaming | 0/5F300B0 | 0/5F300B0 | 0/5F300B0 |
↪| 0/5F2FE48 | 0 | async |
| 3367 | replica2 | 127.0.0.1 | 55589 | streaming | 0/5F300B0 | 0/5F300B0 | 0/5F2FE48 |
↪| 0/5F2FE48 | 0 | async |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
↪+-----+-----+
^C[2015-03-28 21:45:25] INFO: Terminated.
$

```


9.1 Summary

Displays the query and performance information by detecting the PostgreSQL session from capturing the network traffic.

For executing `tcpdump` requires `root` privileges.

9.2 Usage

```
pt-session-profiler [option...]
```

9.3 Options

```
-h, --host=HOSTNAME  
-p, --port=PORT  
-i [INTERFACE]  
-T, --threshold=MILLISECONDS  
--help
```

`-h, --host` Specifies the IP address or PostgreSQL server name of the PostgreSQL session to parse. If not specified, to set the value of `PGHOST`. Every server name or IP address packet will be analyzed if no other value has been defined in `PGHOST`.

`-p, --port` Specifies the port number of the PostgreSQL session to parse. If not specified, to set the value of `PGPORT`. 5432 will be used if no value has been defined in `PGPORT`.

`-i` Specifies the network interface to capture. All network interfaces (`any`) will be used if it is omitted.

`-T, --threshold` Specifies the threshold of execution time in the displayed query. The unit is milliseconds. The default value is 1000.

9.4 Output Items

sess	Unique session strings. (Different values for each session. The source IP address and the port number and the destination IP address and the port number and use the 12 characters of the hash string MD5.)
time	Query execution time
query	Run the query string

9.5 Examples

It monitors the ones destined to port 5432 of the TCP packet that passes through all of the network interface. CTRL-C to end this.

```
$ sudo pt-session-profiler -T 500
[2015-03-29 15:07:22] INFO: Threshold: 500 ms
[2015-03-29 15:07:22] INFO: tcpdump -l -i any -s 0 -X -p tcp port 5432
[2015-03-29 15:07:36] INFO: sess:e27f20dae08f, time:0:00:00.557728, query:UPDATE_
↪pgbench_tellers SET tbalance = tbalance + 2084 WHERE tid = 23;
[2015-03-29 15:07:36] INFO: sess:b3674d7bbea0, time:0:00:00.980950, query:INSERT INTO_
↪pgbench_history (tid, bid, aid, delta, mtime) VALUES (32, 5, 255511, 2695, CURRENT_
↪TIMESTAMP);
[2015-03-29 15:07:45] INFO: sess:1c32286cab7a, time:0:00:01.115904, query:SELECT_
↪abalance FROM pgbench_accounts WHERE aid = 161999;
[2015-03-29 15:07:45] INFO: sess:33f8c268624c, time:0:00:00.526850, query:UPDATE_
↪pgbench_accounts SET abalance = abalance + 3877 WHERE aid = 326415;
[2015-03-29 15:07:46] INFO: sess:b370afd07dcf, time:0:00:00.719780, query:SELECT_
↪abalance FROM pgbench_accounts WHERE aid = 852680;
[2015-03-29 15:07:46] INFO: sess:0f04724051ad, time:0:00:00.543609, query:BEGIN;
^C[2015-03-29 15:07:51] INFO: Terminated.
$
```

10.1 Summary

Change at once the tablespace of the index associated with the specified table.

10.2 Usage

```
pt-set-tablespace [option...] [tablespace]
```

10.3 Options

```
-h, --host=HOSTNAME  
-p, --port=PORT  
-U, --username=USERNAME  
-d, --dbname=DBNAME  
-o, --owner=STRING  
-n, --schema=STRING  
-t, --table=STRING  
-l, --list  
--apply  
--help
```

-h, --host Specifies the connecting PostgreSQL database server name or its IP address. If not specified, to set the value of PGHOST. localhost will be used if no value has been defined in PGHOST.

-p, --port Specifies the port number of the connecting PostgreSQL database. If not specified, to set the value of PGPORT. 5432 will be used if no value has been defined in PGPORT.

-U, --username Specifies the user name of the PostgreSQL database. If not specified, to set the value of PGUSER. The value of USER will be used if no value has been defined in PGUSER.

-d, --dbname Specifies the connecting database name. If not specified, to set the value of PGDATABASE. It connects to the database same as the database username if no value has been defined in PGDATABASE.

-o, --owner It applies only table that is owning user that matches the specified name. Also accepts regular expression when starting and ending with slashes. (cf. /REGEXP/)

-n, --schema It applies only table that is a schema that matches the specified name. Also accepts regular expression when starting and ending with slashes. (cf. /REGEXP/)

-t, --table It applies only table that matches the specified name. Also accepts regular expression when starting and ending with slashes. (cf. /REGEXP/)

--apply The change of tablespace, it is actually reflected in the database.

-l, --list Displays the tablespace information.

-o (or --owner), -n (or --schema), -t (or --table), if specified the options at the same time, the only table that matches the conditions.

If it fails to move one or more tables, returned an exit code of 1. If it's successful the movement of all of the files, returned a "0".

10.4 Output Items

-l, --list Items that are displayed in the options.

OID	Object ID of tablespace
OWNER	Owner name of tablespace
TABLESPACE	Tablespace name
LOCATION	Path of the directory for tablespace
USE%	Disk usage of the partition for tablespace
AVAIL	Free space of the partition for tablespace

Other output items are shown below.

Dry-run mode	Free space of the partition for tablespace.	
Applying ALTER TABLE/ INDEX	Acutually running the ALTER TA- BLE/INDEX statement	it will change tablespace of the index.
X tables/indexes moved. Y failed.	X of tables/indexes is moved successfully	Y failed to move.

10.5 Examples

Displays list of tablespace that exist in the PostgreSQL instance. Gets the used area of the each partition, and displays as a list together.

```
$ pt-set-tablespace --list
+-----+-----+-----+-----+-----+-----+
|  OID   |  OWNER   | TABLESPACE | LOCATION | USE% | AVAIL |
+-----+-----+-----+-----+-----+-----+
|  1663  | postgres | pg_default  |          |     |      |
|  1664  | postgres | pg_global   |          |     |      |
| 121263 | postgres | hddspc2    | /disk/disk2/pgsql | 85% | 80G  |
```

(continues on next page)

(continued from previous page)

```

| 16818 | postgres | ssdspc1 | /disk/disk1/tblspc1 | 67% | 127G |
| 305242 | postgres | ssdspc2 | /disk/disk3/pgsql/ssdspc2 | 98% | 13G |
+-----+-----+-----+-----+-----+-----+
$

```

All orders tables and the indexes that was created on orders table in dbt3 database, it displays ALTER TABLE and ALTER INDEX statement for to move the tablespace. (Does not actually move)

```

$ pt-set-tablespace -d dbt3 --table orders ssdspc1
[2015-04-29 17:35:24] INFO: Dry-run mode:
[2015-04-29 17:35:24] INFO: ALTER TABLE "public"."orders" SET TABLESPACE "ssdspc1";
[2015-04-29 17:35:24] INFO: ALTER INDEX "public"."pk_orders" SET TABLESPACE "ssdspc1";
[2015-04-29 17:35:24] INFO: ALTER INDEX "public"."i_o_orderdate" SET TABLESPACE
↪ "ssdspc1";
[2015-04-29 17:35:24] INFO: ALTER INDEX "public"."i_o_custkey" SET TABLESPACE "ssdspc1
↪ ";
$

```

All orders tables and the indexes that were created on orders table in dbt3 database, to move to the ssdspc1 tablespace.

```

$ pt-set-tablespace -d dbt3 --table orders --apply ssdspc1
[2015-04-29 17:37:06] INFO: Applying ALTER TABLE/INDEX...
[2015-04-29 17:37:08] INFO: 4 tables/indexes moved. 0 failed.
$

```

All tables in the dbt3 schema and all indexes, to move to the ssdspc1 tablespace.

```

$ pt-set-tablespace --schema dbt3 --apply ssdspc1
[2015-04-29 17:38:39] INFO: Applying ALTER TABLE/INDEX...
[2015-04-29 17:38:57] INFO: 31 tables/indexes moved. 0 failed.
$

```


11.1 Summary

Shows waiting and blocking lock information and those dependencies with the associated queries.

11.2 Usage

```
pt-show-locks [option...]
```

11.3 Options

```
-h, --host=HOSTNAME  
-p, --port=PORT  
-U, --username=USERNAME  
-d, --dbname=DBNAME  
--help
```

-h, --host Specifies the connecting the PostgreSQL database server name or its IP address. If not specified, to set the value of PGHOST. `localhost` will be used if no other value has been defined in PGHOST.

-p, --port Specifies the port number of the connecting PostgreSQL database. If not specified, to set the value of PGPORT. `5432` will be used if no value has been defined in PGPORT.

-U, --username Specifies the user name of the PostgreSQL database. If not specified, to set the value of PGUSER. The value of `USER` will be used If no other value has been defined in PGUSER.

-d, --dbname Specifies the connecting database name. If not specified, to set the value of PGDATABASE. Database name as database user name will be used if no other value has been defined in PGDATABASE.

11.4 Output Items

BLOCKER	Blocking process id when the lock is waiting.	
PID	Backend process id.	
LOCKTYPE	Lock type.	
MODE	Lock mode.	
GRANTED	t on granted	f on waiting.
QUERY	Query string being executed.	

11.5 Examples

Following example shows the lock status with connecting mydb database with postgres user.

```
$ pg-show-locks -d mydb -U postgres
+-----+-----+-----+-----+-----+-----+
↪-----+
| BLOCKER | PID | LOCKTYPE | MODE | GRANTED | QUERY_
↪
+-----+-----+-----+-----+-----+-----+
↪-----+
|          | 2076 | relation | AccessExclusiveLock | t | lock TABLE pgbench_
↪accounts; |
| 2076 | 2049 | relation | RowShareLock | f | select * from pgbench_
↪accounts for update; |
| 2076 | 2436 | relation | RowExclusiveLock | f | delete from pgbench_
↪accounts; |
+-----+-----+-----+-----+-----+-----+
↪-----+
$
```


12.1 Summary

Displays by calculating the difference between the statistics of two difference time of SQL statements.

If it has specified options value, it can be displayed to sort of each specific items.

Make sure you have `pg_stat_statements` of contrib module.

And also make sure to enable `track_io_timing` option.

12.2 Usage

```
pt-snap-statements [option...] [interval]
```

12.3 Options

```
-h, --host=HOSTNAME  
-p, --port=PORT  
-U, --username=USERNAME  
-d, --dbname=DBNAME  
-s, --sort=KEY  
-l  
-t, --top=NUMBER  
-R, --reset  
--help
```

`-h, --host` Specifies the connecting the PostgreSQL database server name or its IP address. If not specified, to set the value of `PGHOST`. “localhost” will be used if no other value has been defined in `PGHOST`.

-p, --port Specifies the port number of the connecting the PostgreSQL database. If not specified, to set the value of PGPORT. 5432 will be used if no value has been defined in PGPORT.

-U, --username Specifies the user name of the PostgreSQL database. If not specified, to set the value of PGUSER. Vars USER will be used if no other value has been defined in PGUSER.

-d, --dbname Specifies the connecting database name. If not specified, to set the value of PGDATABASE. Database name as database user name will be used if no other value has been defined in PGDATABASE.

-s (not implemented) Specifies the sort item. KEY can take one of following values: CALLS, T_TIME, ROWS, B_HIT, B_READ, B_DIRT, B_WRTN, R_TIME, W_TIME

-l (not implemented) Displays with detailed every block classification (Shared buffer, Local buffer, Temporary buffer). If not specified, it is displayed the total number of shared buffer, local buffer and temporary buffer.

-t, --top Specifies the number of displayed queries. If not specified, it is displayed the all queries.

-R, --reset It initializes the statistical information of ‘pg_stat_statements‘ view.

12.4 Output Items

USER	Username of query run
DBNAME	Database name of query run
QUERYID	Query ID of query run (Hexadecimal)
QUERY	Query performed (Display up to 30 characters)
CALLS	Number of times of query run
T_TIME	Total number of times of query run (Millisecond)
ROWS	Total number of rows that has received the obtain or influence
B_HIT	Total number of blocks read from the buffer at the time blocks read
B_READ	Total number of blocks read from the disk at the time blocks read
B_DIRT	Total number of pages that page has been updated by the query
B_WRTN	Total number of blocks that are written to disk by the query
R_TIME	Total time of block read from the disk (Millisecond) (Make sure to enable track_io_timing parameter)
W_TIME	Total time of block write to the disk (Millisecond) (Make sure to enable track_io_timing parameter)

12.5 Examples

Connects to the postgres database, and the SQL statements executed in 5 seconds is sorted in descending order of total execution time (T_TIME), and displays all.

```
$ pt-snap-statements -d postgres 5
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
↪+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| USER | DBNAME | QUERYID |          QUERY          | CALLS | T_TIME | ↪
↪ROWS | B_HIT | B_READ | B_DIRT | B_WRTN | R_TIME | W_TIME |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
↪+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| snaga | postgres | 80053daf | UPDATE pgbench_branches SET bb | 677 | 12007 | ↪
↪677 | 9160 | 1 | 1 | 0 | 0.0 | 0.0 |
| snaga | postgres | 1675159e | UPDATE pgbench_tellers SET tba | 681 | 7648 | ↪
↪681 | 3403 | 0 | 0 | 0 | 0.0 | 0.0 |
```

(continues on next page)

(continued from previous page)

snaga postgres ec088219 UPDATE pgbench_accounts SET ab	684	530	↳
↳ 684 2289 585 568	0 125.9 0.0		
snaga postgres 198383d SELECT abalance FROM pgbench_a	682	73	↳
↳ 682 2080 0 0	0 0.0 0.0		
snaga postgres da8cc6f INSERT INTO pgbench_history (t	676	34	↳
↳ 676 704 12 10	0 0.0 0.0		
snaga postgres d4e6bf94 BEGIN;	684	4	↳
↳ 0 0 0 0	0 0.0 0.0		
snaga postgres a81672e END;	671	3	↳
↳ 0 0 0 0	0 0.0 0.0		
snaga postgres 8caa574 select count(*) from pgbench_b	1	0	↳
↳ 1 4 0 0	0 0.0 0.0		
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+			
↳ +-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+			
\$			

User snaga connects to the postgres database of the PostgreSQL server running on the port 5433 of host 192.168.1.101, and the SQL statements executed in 5 seconds is sorted in descending order of total execution time (T_TIME), and the top 5 are displayed.

```
$ pt-snap-statements --host 192.168.1.101 -p 5433 -U snaga -d postgres -t 5 5
```

USER	DBNAME	QUERYID	QUERY	CALLS	T_TIME	↳
ROWS	B_HIT	B_READ	B_DIRT	B_WRTN	R_TIME	W_TIME
snaga postgres 80053daf UPDATE pgbench_branches SET bb	503	9953	↳			
↳ 503 8430 14 7	0 0.6 0.0					
snaga postgres 1675159e UPDATE pgbench_tellers SET tba	508	6483	↳			
↳ 508 2551 10 9	0 0.3 0.0					
snaga postgres ec088219 UPDATE pgbench_accounts SET ab	511	560	↳			
↳ 511 1424 698 477	7 91.0 12.1					
snaga postgres 198383d SELECT abalance FROM pgbench_a	511	93	↳			
↳ 511 1550 0 0	0 0.0 0.0					
snaga postgres da8cc6f INSERT INTO pgbench_history (t	503	20	↳			
↳ 503 530 13 11	0 0.1 0.0					
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+						
↳ +-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+						
\$						

13.1 Summary

Get a snapshot of statistical information, save, and manage.

13.2 Usage

```
pt-stat-snapshot [option...] install
pt-stat-snapshot [option...] uninstall
pt-stat-snapshot [option...] create [level]
pt-stat-snapshot [option...] list
pt-stat-snapshot [option...] delete [sid]
pt-stat-snapshot [option...] export [file]
pt-stat-snapshot [option...] import [file]
```

13.3 Commands

install	Creates a schema and tables and functions of the package.
uninstall	Drops a schema and tables and functions of the package.
create [level]	Takes a snapshot of the database statistics. [level] can be 1 or 2 or 4.
list	Shows a list of stored snapshots.
delete [sid]	Deletes a snapshot specified by snapshot id. Multiple snapshot ids can be specified by range as M:N.
export [file]	Exports all snapshot data into the specified file.
import [file]	Imports snapshot data from the specified file.

13.4 Options

```
-h, --host=HOSTNAME
-p, --port=PORT
-U, --username=USERNAME
-d, --dbname=DBNAME
--help
```

`-h, --host` Specifies the connecting the PostgreSQL database server name or its IP address. or its IP address. If not specified, to set the value of `PGHOST`. `localhost` will be used if no other value has been defined in `PGHOST`.

`-p, --port` Specifies the port number of the connecting PostgreSQL database. If not specified, to set the value of `PGPORT`. `5432` will be used if no value has been defined in `PGPORT`.

`-U, --username` Specifies the username of the PostgreSQL database. If not specified, to set the value of `PGUSER`. The value of `USER` will be used If no other value has been defined in `PGUSER`.

`-d, --dbname` Specifies the connecting database name. If not specified, to set the value of `PGDATABASE`. Database name as database username will be used if no other value has been defined in `PGDATABASE`.

13.5 Output Items

`list` command shows following items.

SID	Snapshot ID (Assigned to every snapshot. Monotonic increase.)
TIMESTAMP	Timestamp of the snapshot taken.
LEVEL	Snapshot level.

13.6 Examples

It installs a schema, tables and functions of the package to `testdb` database.

```
$ pt-stat-snapshot -h 127.0.0.1 -U postgres -d testdb install
[2015-03-31 17:21:37] INFO: Succeeded to install ppperf snapshot.
$
```

It takes a snapshot with the snapshot level 4.

```
$ pt-stat-snapshot -h 127.0.0.1 -U postgres -d testdb create 4
[2015-03-31 17:21:47] INFO: Succeeded to take a snapshot.
$
```

It shows a list of the snapshots.

```
$ pt-stat-snapshot -h 127.0.0.1 -U postgres -d testdb list
+-----+-----+-----+
| SID |      TIMESTAMP      | LEVEL |
+-----+-----+-----+
|  0  | 2015-03-31 17:21:47 |    1  |
+-----+-----+-----+
$
```

It uninstalls the schema, tables and functions of the package from `testdb` database.

```
$ pt-stat-snapshot -h 127.0.0.1 -U postgres -d testdb uninstall
[2015-03-31 17:21:59] INFO: Succeeded to uninstall pgperf snapshot.
$
```

13.7 About pgperf-snapshot Module

`pt-stat-snapshot` command depends on another module internally, previously known as the `pgperf-snapshot`.

For more details about the `pgperf-snapshot` module, please refer to the following documents.

13.7.1 PgPerf Package Overview

What is PgPerf package?

The PgPerf package is a collection of SQL functions and tables to take snapshots of the performance statistics in PostgreSQL, and store them into dedicated snapshot tables.

“Performance statistics” mentioned here is following statistics available in PostgreSQL:

- Database object access statistics in system views. (c.f. `pg_stat_database`)
- Optimizer statistics in `pg_statistic` system table.
- Other system statistics which could be taken by system functions. (c.f. `pg_current_xlog_location()`)
- Fragmentation statistics which could be taken by system functions. (c.f. `pgstattuple()`)

By taking those performance statistics and storing them, DBA can understand what’s actually going on inside the PostgreSQL databases, and also it would help DBA predict the future trend in terms of the performance.

Advantages of PgPerf package

The PgPerf package has following advantages:

- It does not depend the platform where PostgreSQL is running. Provided scripts (SQL, PL/pgSQL) should work wherever you want.
- It’s easy to take and store several performance metrics, and it’s also easy to analyze them later whatever you want.
- It’s easy to deploy (and undeploy) without changing any configuration in the production environment.

Supported Platform

The PgPerf package would work with following platforms.

- PostgreSQL 9.0, 9.1, 9.2, 9.3, 9.4

It does not depend on the operating system.

PgPerf Components

The PgPerf package consists of following components:

- Dedicated schema to be used by the package.
- SQL functions to take performance snapshots.
- Snapshot tables to store performance snapshots.
- Some shell scripts for convenience.

13.7.2 Introduction to PgPerf package

This chapter is intended to give a brief introduction to the PgPerf snapshot package.

Installing PgPerf package

To install the PgPerf package, `pgperf_snapshot_install.sql` script needs to be executed on a database, where the DBA wants to take performance snapshots, to create `pgperf` schema, snapshot functions, and snapshot tables.

```
psql -f pgperf_snapshot_install<VERSION>.sql <DBNAME>
```

Uninstalling PgPerf package

To uninstall the PgPerf package, `pgperf_snapshot_uninstall.sql` script needs to be executed on the database where the PgPerf package has been installed.

```
psql -f pgperf_snapshot_uninstall.sql <DBNAME>
```

Once `pgperf_snapshot_uninstall.sql` is executed on a database, it would drop the `pgperf` schema, snapshot functions and snapshot tables installed in the database.

Taking a performance snapshot with using PgPerf package

In order to take a performance snapshot, a SQL function `pgperf.create_snapshot()` needs to be called.

Once `pgperf.create_snapshot()` function is called with specifying a snapshot level, a snapshot of the several performance statistics would be taken, and a snapshot id associated with the performance snapshot would be returned.

In the following example, a snapshot level is specified as 4, and then, 1005 is returned as the snapshot id.

```
postgres=# SELECT pgperf.create_snapshot(4);
 create_snapshot
-----
                1005
(1 row)

postgres=#
```


Showing a list for snapshots

`pgperf.snapshot` table would hold a list of the snapshots.

By querying the table, snapshot ids and those timestamps can be obtained as following.

```
postgres=# SELECT * FROM pgperf.snapshot;
 sid |          ts          | level
-----+-----+-----
   0 | 2015-04-11 19:11:24.04428 |    1
   1 | 2015-04-11 19:11:24.060965 |    2
   2 | 2015-04-11 19:11:24.110034 |    4
(3 rows)

postgres=#
```

Deleting a snapshot

To delete a snapshot, `pgperf.delete_snapshot()` function with specifying a snapshot id can be used.

In the following example, a snapshot associated with snapshot id 2 is going to be dropped.

```
postgres=# SELECT pgperf.delete_snapshot(2);
 delete_snapshot
-----
 t
(1 row)

postgres=# SELECT * FROM pgperf.snapshot;
 sid |          ts          | level
-----+-----+-----
   0 | 2015-04-11 19:11:24.04428 |    1
   1 | 2015-04-11 19:11:24.060965 |    2
(2 rows)

postgres=#
```

Taking snapshots as a routine

To take a snapshot as a routine, `pgperf.create_snapshot()` function needs to be called with using a cron-like tool.

A shell script, `get_snapshot.sh`, which is contained in the package, executes `pgperf.create_snapshot()` function on the specified database or all databases, which can be connected by the script (and not a template database).

In the following example, the script is taking a snapshot for `postgres` database.

```
$ ./get_snapshot.sh postgres
```

On the other hand, the script is taking a snapshot for every database in the following example.

```
$ ./get_snapshot.sh
```

By setting a crontab, `get_snapshot.sh` can be called to take snapshots periodically.

A crontab shown in below is intended to take performance snapshots of all the databases with calling `get_snapshot.sh` script in every 10 minutes.

```
0-59/10 * * * * /path/to/get_snapshot.sh > /dev/null 2>&1
```

Purging snapshots

`pgperf.purge_snapshots()` function can be used to purge older snapshots at once.

In the following example, snapshots, which are older than 1 week or more, are going to be purged.

```
postgres=# SELECT sid,ts FROM pgperf.snapshot ORDER BY ts LIMIT 1;
 sid |          ts
-----+-----
   2 | 2012-10-21 18:20:01.238885
(1 row)

postgres=# SELECT now(),pgperf.purge_snapshots('1 weeks');
          now          | purge_snapshots
-----+-----
2012-10-29 14:57:04.092243+09 |          121
(1 row)

postgres=# SELECT sid,ts FROM pgperf.snapshot ORDER BY ts LIMIT 1;
 sid |          ts
-----+-----
 123 | 2012-10-22 15:00:01.8397
(1 row)

postgres=#
```

13.7.3 Snapshot Functions

This chapter is intended to give detailed information about several snapshot functions, provided in the PgPerf package, to deal with performance snapshots.

Snapshot Function List

Function Name	Description
<code>pgperf.create_snapshot(level)</code>	Takes a new snapshot.
<code>pgperf.delete_snapshot(snapid)</code>	Drops a snapshot specified by the snapshot id.
<code>pgperf.purge_snapshots(interval)</code>	Drops snapshots older than the specified period.
<code>pgperf.get_interval(snapid1, snapid2)</code>	Gets an interval (in second) between two snapshots.

`pgperf.create_snapshot()` Function

Description

`pgperf.create_snapshot()` function takes a snapshot of the performance statistics, which can be obtained in the PostgreSQL database, and stores them in the snapshot tables.

Declaration

```
integer pgperf.create_snapshot(integer level)
```

Parameters

Name	Type	Description
level	integer	Snapshot level to be taken.

Unfortunately, heavy performance impact could sometimes be generated while taking a snapshot for all the performance statistics available in the database.

To avoid that, `pgperf.create_snapshot()` function allows DBA to obtain a snapshot with less performance impact by specifying the snapshot level. Then, DBA can avoid taking a snapshot which generates heavy performance impact so frequently.

Snapshot Level	Snapshot Contents
1	Obtains a snapshot of the basic access statistics and the session statistics. <code>pg_stat_database</code> , <code>pg_database_size()</code> <code>pg_stat_user_tables</code> , <code>pg_statio_user_tables</code> <code>pg_stat_user_indexes</code> , <code>pg_statio_user_indexes</code> <code>pg_relation_size()</code> , <code>pg_total_relation_size()</code> <code>pg_current_xlog_location()</code> , <code>pg_current_xlog_insert_location()</code> <code>pg_stat_bgwriter</code> <code>pg_stat_activity</code> , <code>pg_locks</code> , <code>pg_stat_statements</code>
2	In addition to the level 1, obtains a snapshot of the optimizer statistics. <code>pg_statistic</code>
3	Not used.
4	In addition to the level 2, obtains a snapshot of the table/index fragmentation statistics. <code>pgstattuple()</code> , <code>pgstatindex()</code>
5	Not used.

pgperf.delete_snapshot() Function

Description

`pgperf.delete_snapshot()` function deletes a snapshot of the performance statistics specified by the snapshot id.

Declaration

```
integer pgperf.delete_snapshot(integer snapid);
```

Parameters

Name	Type	Description
snapid	integer	A snapshot id associated with the snapshot to be deleted.

pgperf.purge_snapshots() Function

Description

`pgperf.purge_snapshots()` function purges older performance snapshots at once.

Declaration

```
integer pgperf.purge_snapshots(interval period);
```

Parameters

Name	Type	Description
period	interval	A period of the snapshots to be kept in the snapshot tables.

This function drops snapshots older than the period (interval) specified by the parameter `period`.

See the PostgreSQL manual for more details about how to express an `interval` value.

- [PostgreSQL: Documentation: 9.0: Date/Time Types](#)

pgperf.get_interval() Function

Description

`pgperf.get_interval()` function obtains an interval between two snapshots.

This function is intended to give some convenience to convert an obtained value into a ‘per-second’ value, particularly in SQL scripts.

Declaration

```
integer pgperf.get_interval(integer snapid1, integer snapid2)
```

Parameters

Name	Type	Description
snapid1	integer	A snapshot id at the start point.
snapid2	integer	A snapshot id at the end point.

13.7.4 Snapshot Tables

This chapter is intended to give detailed information about several snapshot tables, where the performance snapshot to be stored.

Snapshot Table List

Performance snapshot is going to be stored in the following tables associated with each performance statistics available in PostgreSQL.

Table Name	Description	Note
pgperf.snapshot	Stores snapshot id and timestamp of the performance snapshot.	
pgperf.snapshot_pg_stat_database	Stores a snapshot of the pg_stat_database system view.	
pgperf.snapshot_pg_database_size	Stores a snapshot of the database size.	
pg-perf.snapshot_pg_stat_user_tables	Stores a snapshot of the pg_stat_user_tables system view.	
pg-perf.snapshot_pg_statio_user_tables	Stores a snapshot of the pg_statio_user_tables system view.	
pg-perf.snapshot_pg_stat_user_indexes	Stores a snapshot of the pg_stat_user_indexes system view.	
pg-perf.snapshot_pg_statio_user_indexes	Stores a snapshot of the pg_statio_user_indexes system view.	
pg-perf.snapshot_pg_statio_user_sequences	Stores a snapshot of the pg_statio_user_sequences system view.	
pg-perf.snapshot_pg_stat_user_functions	Stores a snapshot of the pg_stat_user_functions system view.	
pgperf.snapshot_pg_relation_size	Stores a snapshot of the table and index size.	
pgperf.snapshot_pg_current_xlog	Stores a snapshot of the current xlog location and the current insert location.	
pgperf.snapshot_pg_stat_bgwriter	Stores a snapshot of the pg_stat_bgwriter system view.	
pgperf.snapshot_pg_stat_activity	Stores a snapshot of the pg_stat_activity system view.	
pgperf.snapshot_pg_locks	Stores a snapshot of the pg_locks system view.	
pgperf.snapshot_pg_statistic	Stores a snapshot of the pg_statistics system table.	
pg-perf.snapshot_pg_stat_statements	Stores a snapshot of the pg_stat_statements view.	8.4 or later
pgperf.snapshot_pgstattuple	Stores a snapshot of the result of pgstattuple function.	
pgperf.snapshot_pgstatindex	Stores a snapshot of the result of pgstatindex function.	

pgperf.snapshot Table

This table stores snapshot id and timestamp of each snapshot taken by the snapshot function.

Column Name	Type	Source	Note
sid	integer	Snapshot ID	Monotone increasing
ts	timestamp	Timestamp of the snapshot	
level	integer	Snapshot level	

pgperf.snapshot_pg_stat_database Table

This table stores snapshots of the pg_stat_database system view which contains the database access statistics.

Column Name	Type	Source	Note
sid	integer	Snapshot ID	
datid	oid	pg_stat_database.datid	
datname	name	pg_stat_database.datname	
numbackends	integer	pg_stat_database.numbackends	
xact_commit	bigint	pg_stat_database.xact_commit	
xact_rollback	bigint	pg_stat_database.xact_rollback	
blks_read	bigint	pg_stat_database.blks_read	
blks_hit	bigint	pg_stat_database.blks_hit	
tup_returned	bigint	pg_stat_database.tup_returned	
tup_fetched	bigint	pg_stat_database.tup_fetched	
tup_inserted	bigint	pg_stat_database.tup_inserted	
tup_updated	bigint	pg_stat_database.tup_updated	
tup_deleted	bigint	pg_stat_database.tup_deleted	
conflicts	bigint	pg_stat_database.conflicts	9.1 or later
stats_reset	timestampz	pg_stat_database.stats_reset	9.1 or later

pgperf.snapshot_pg_database_size Table

This table stores snapshots of result of the `pg_database_size()` function which gets database size.

Column Name	Type	Source	Note
sid	integer	Snapshot ID	
datname	name	pg_database.datname	
pg_database_size	bigint	pg_database_size()	

pgperf.snapshot_pg_stat_user_tables Table

This table stores snapshots of the `pg_stat_user_tables` system view which contains the table access statistics.

Column Name	Type	Source	Note
sid	integer	Snapshot ID	
relid	oid	pg_stat_user_tables.relid	
schemaname	name	pg_stat_user_tables.schemaname	
relname	name	pg_stat_user_tables.relname	
seq_scan	bigint	pg_stat_user_tables.seq_scan	
seq_tup_read	bigint	pg_stat_user_tables.seq_tup_read	
idx_scan	bigint	pg_stat_user_tables.idx_scan	
idx_tup_fetch	bigint	pg_stat_user_tables.idx_tup_fetch	
n_tup_ins	bigint	pg_stat_user_tables.n_tup_ins	
n_tup_upd	bigint	pg_stat_user_tables.n_tup_upd	
n_tup_del	bigint	pg_stat_user_tables.n_tup_del	
n_tup_hot_upd	bigint	pg_stat_user_tables.n_tup_hot_upd	
n_live_tup	bigint	pg_stat_user_tables.n_live_tup	
n_dead_tup	bigint	pg_stat_user_tables.n_dead_tup	
last_vacuum	timestampz	pg_stat_user_tables.last_vacuum	
last_autovacuum	timestampz	pg_stat_user_tables.last_autovacuum	
last_analyze	timestampz	pg_stat_user_tables.last_analyze	
last_autoanalyze	timestampz	pg_stat_user_tables.last_autoanalyze	
vacuum_count	bigint	pg_stat_user_tables.vacuum_count	9.1 or later
autovacuum_count	bigint	pg_stat_user_tables.autovacuum_count	9.1 or later
analyze_count	bigint	pg_stat_user_tables.analyze_count	9.1 or later
autoanalyze_count	bigint	pg_stat_user_tables.autoanalyze_count	9.1 or later

pgperf.snapshot_pg_statio_user_tables Table

This table stores snapshots of the `pg_statio_user_tables` system view which contains the table access statistics.

Column Name	Type	Source	Note
sid	integer	Snapshot ID	
relid	oid	pg_statio_user_tables.relid	
schemaname	name	pg_statio_user_tables.schemaname	
relname	name	pg_statio_user_tables.relname	
heap_blks_read	bigint	pg_statio_user_tables.heap_blks_read	
heap_blks_hit	bigint	pg_statio_user_tables.heap_blks_hit	
idx_blks_read	bigint	pg_statio_user_tables.idx_blks_read	
idx_blks_hit	bigint	pg_statio_user_tables.idx_blks_hit	
toast_blks_read	bigint	pg_statio_user_tables.toast_blks_read	
toast_blks_hit	bigint	pg_statio_user_tables.toast_blks_hit	
tidx_blks_read	bigint	pg_statio_user_tables.tidx_blks_read	
tidx_blks_hit	bigint	pg_statio_user_tables.tidx_blks_hit	

pgperf.snapshot_pg_stat_user_indexes Table

This table stores snapshots of the `pg_stat_user_indexes` system view which contains the index access statistics.

Column Name	Type	Source	Note
sid	integer	Snapshot ID	
relid	oid	pg_stat_user_indexes.relid	
indexrelid	oid	pg_stat_user_indexes.indexrelid	
schemaname	name	pg_stat_user_indexes.schemaname	
relname	name	pg_stat_user_indexes.relname	
indexrelname	name	pg_stat_user_indexes.indexrelname	
idx_scan	bigint	pg_stat_user_indexes.idx_scan	
idx_tup_read	bigint	pg_stat_user_indexes.idx_tup_read	
idx_tup_fetch	bigint	pg_stat_user_indexes.idx_tup_fetch	

pgperf.snapshot_pg_statio_user_indexes Table

This table stores snapshots of the `pg_statio_user_indexes` system view which contains the index access statistics.

Column Name	Type	Source	Note
sid	integer	Snapshot ID	
relid	oid	pg_statio_user_indexes.relid	
indexrelid	oid	pg_statio_user_indexes.indexrelid	
schemaname	name	pg_statio_user_indexes.schemaname	
relname	name	pg_statio_user_indexes.relname	
indexrelname	name	pg_statio_user_indexes.indexrelname	
idx_blks_read	bigint	pg_statio_user_indexes.idx_blks_read	
idx_blks_hit	bigint	pg_statio_user_indexes.idx_blks_hit	

pgperf.snapshot_pg_statio_user_sequences Table

This table stores snapshots of the `pg_statio_user_sequences` system view which contains the sequence access statistics.

Column Name	Type	Source	Note
sid	integer	Snapshot ID	
relid	oid	pg_statio_user_sequences.relid	
schemaname	name	pg_statio_user_sequences.schemaname	
relname	name	pg_statio_user_sequences.relname	
blks_read	int8	pg_statio_user_sequences.blks_read	
blks_hit	int8	pg_statio_user_sequences.blks_hit	

pgperf.snapshot_pg_stat_user_functions Table

This table stores snapshots of the `pg_stat_user_functions` system view which contains the function access statistics.

Column Name	Type	Source	Note
sid	integer	Snapshot ID	
funcid	oid	pg_stat_user_functions.funcid	
schemaname	name	pg_stat_user_functions.schemaname	
funcname	name	pg_stat_user_functions.funcname	
calls	int8	pg_stat_user_functions.calls	
total_time	int8	pg_stat_user_functions.total_time	
self_time	int8	pg_stat_user_functions.self_time	

pgperf.snapshot_pg_relation_size Table

This table stores snapshots of the result of `pg_relation_size()` and `pg_total_relation_size()` function which gets table and/or index size.

Column Name	Type	Source	Note
sid	integer	Snapshot ID	
schemaname	name	pg_stat_user_tables.schemaname, pg_stat_user_indexes.schemaname	
relid	oid	pg_stat_user_tables.relid, pg_stat_user_indexes.indexrelid	
relname	name	pg_class.relname	
pg_relation_size	bigint	pg_relaion_size()	
pg_total_relation_size	bigint	pg_total_relaion_size()	Available only for tables

pgperf.snapshot_pg_current_xlog Table

This table stores snapshots of the result of `pg_current_xlog_location()` and `pg_current_xlog_insert_location()` function which gets current WAL locations.

Column Name	Type	Source	Note
sid	integer	Snaphsot ID	
location	text	pg_current_xlog_location()	
insert_location	text	pg_current_xlog_insert_location()	

pgperf.snapshot_pg_stat_bgwriter Table

This table stores snapshots of the `pg_stat_bgwriter` system view which contains the background writer statistics.

Column Name	Type	Source	Note
sid	integer	Snapshot ID	
checkpoints_timed	bigint	pg_stat_bgwriter.checkpoints_timed	
checkpoints_req	bigint	pg_stat_bgwriter.checkpoints_req	
checkpoint_write_time	double precision	pg_stat_bgwriter.checkpoint_write_time	9.2 or later
checkpoint_sync_time	double precision	pg_stat_bgwriter.checkpoint_sync_time	9.2 or later
buffers_checkpoint	bigint	pg_stat_bgwriter.buffers_checkpoint	
buffers_clean	bigint	pg_stat_bgwriter.buffers_clean	
maxwritten_clean	bigint	pg_stat_bgwriter.maxwritten_clean	
buffers_backend	bigint	pg_stat_bgwriter.buffers_backend	
buffers_backend_fsync	bigint	pg_stat_bgwriter.buffers_backend_fsync	9.1 or later
buffers_alloc	bigint	pg_stat_bgwriter.buffers_alloc	
stats_reset	timestampz	pg_stat_bgwriter.stats_reset	9.1 or later

pgperf.snapshot_pg_stat_activity Table

This table stores snapshots of the `pg_stat_activity` system view which contains the session information.

Column Name	Type	Source	Note
sid	integer	Snapshot ID	
datid	oid	pg_stat_activity.datid	
datname	name	pg_stat_activity.datname	
procpid	int4	pg_stat_activity.procpid	9.1 or before
pid	int4	pg_stat_activity.pid	9.2 or later
usesysid	oid	pg_stat_activity.usesysid	
username	name	pg_stat_activity.username	
application_name	text	pg_stat_activity.application_name	9.0 or later
client_addr	inet	pg_stat_activity.client_addr	
client_hostname	text	pg_stat_activity.client_hostname	9.1 or later
client_port	int4	pg_stat_activity.client_port	
backend_start	timestampz	pg_stat_activity.backend_start	
xact_start	timestampz	pg_stat_activity.xact_start	
query_start	timestampz	pg_stat_activity.query_start	
state_change	timestampz	pg_stat_activity.state_change	9.2 or later
waiting	bool	pg_stat_activity.waiting	
state	text	pg_stat_activity.state	9.2 or later
current_query	text	pg_stat_activity.current_query	9.1 or before
query	text	pg_stat_activity.query	9.2 or later

pgperf.snapshot_pg_locks Table

This table stores snapshots of the `pg_locks` system view which contains the lock information.

Column Name	Type	Source	Note
sid	integer	Snapshot ID	
locktype	text	pg_locks.locktype	
database	oid	pg_locks.database	
relation	oid	pg_locks.relation	
page	int4	pg_locks.page	
tuple	int2	pg_locks.tuple	
virtualxid	text	pg_locks.virtualxid	
transactionid	xid	pg_locks.transactionid	
classid	oid	pg_locks.classid	
objid	oid	pg_locks.objid	
objsubid	int2	pg_locks.objsubid	
virtualtransaction	text	pg_locks.virtualtransaction	
pid	int4	pg_locks.pid	
mode	text	pg_locks.mode	
granted	bool	pg_locks.granted	
fastpath	bool	pg_locks.fastpath	9.2 or later

pgperf.snapshot_pg_statistic Table

This table stores snapshots of the `pg_statistic` system table which contains the optimizer statistics.

Column Name	Type	Source	Note
sid	integer	Snapshot ID	
starelid	oid	pg_statistic.starelid	
starelname	name	pg_class.relname	
staattnum	smallint	pg_statistic.staattnum	
staattname	name	pg_attribute.attname	
stainherit	boolean	pg_statistic.stainherit	9.0 or later
stanullfrac	real	pg_statistic.stanullfrac	
stawidth	integer	pg_statistic.stawidth	
stadistinct	real	pg_statistic.stadistinct	
stakind1	smallint	pg_statistic.stakind1	
stakind2	smallint	pg_statistic.stakind2	
stakind3	smallint	pg_statistic.stakind3	
stakind4	smallint	pg_statistic.stakind4	
stakind5	smallint	pg_statistic.stakind5	9.2 or later
staop1	oid	pg_statistic.staop1	
staop2	oid	pg_statistic.staop2	
staop3	oid	pg_statistic.staop3	
staop4	oid	pg_statistic.staop4	
staop5	oid	pg_statistic.staop5	9.2 or later
stanumbers1	real[]	pg_statistic.stanumbers1	
stanumbers2	real[]	pg_statistic.stanumbers2	
stanumbers3	real[]	pg_statistic.stanumbers3	
stanumbers4	real[]	pg_statistic.stanumbers4	
stanumbers5	real[]	pg_statistic.stanumbers5	9.2 or later
stavalues1	text	pg_statistic.stavalues1	
stavalues2	text	pg_statistic.stavalues2	
stavalues3	text	pg_statistic.stavalues3	
stavalues4	text	pg_statistic.stavalues4	
stavalues5	text	pg_statistic.stavalues5	9.2 or later

pgperf.snapshot_pg_stat_statements Table

This table stores snapshots of the `pg_stat_statements` view which contains the session statistics. This table is available only when the `pg_stat_statements` module has been installed and enabled.

Column Name	Type	Source	Note
sid	integer	Snapshot ID	
userid	oid	pg_stat_statements.userid	
dbid	oid	pg_stat_statements.dbid	
query	text	pg_stat_statements.query	
calls	bigint	pg_stat_statements.calls	
total_time	double precision	pg_stat_statements.total_time	
rows	bigint	pg_stat_statements.rows	
shared_blks_hit	bigint	pg_stat_statements.shared_blks_hit	
shared_blks_read	bigint	pg_stat_statements.shared_blks_read	
shared_blks_dirtied	bigint	pg_stat_statements.shared_blks_dirtied	9.2 or later
shared_blks_written	bigint	pg_stat_statements.shared_blks_written	
local_blks_hit	bigint	pg_stat_statements.local_blks_hit	
local_blks_read	bigint	pg_stat_statements.local_blks_read	
local_blks_dirtied	bigint	pg_stat_statements.local_blks_dirtied	9.2 or later
local_blks_written	bigint	pg_stat_statements.local_blks_written	
temp_blks_read	bigint	pg_stat_statements.temp_blks_read	
temp_blks_written	bigint	pg_stat_statements.temp_blks_written	
blk_read_time	double precision	pg_stat_statements.blk_read_time	9.2 or later
blk_write_time	double precision	pg_stat_statements.blk_write_time	9.2 or later

pgperf.snapshot_pgstattuple Table

This table stores snapshots of the result of the `pgstattuple()` function which gets the table fragmentation statistics. This table is available only when the `pgstattuple` module has been installed.

Column Name	Type	Source	Note
sid	integer	Snapshot ID	
schemaname	name	pg_stat_user_tables.schemaname	
relname	name	pg_stat_user_tables.relname	
table_len	int8	pgstattuple().table_len	
tuple_count	int8	pgstattuple().tuple_count	
tuple_len	int8	pgstattuple().tuple_len	
tuple_percent	float8	pgstattuple().tuple_percent	
dead_tuple_count	int8	pgstattuple().dead_tuple_count	
dead_tuple_len	int8	pgstattuple().dead_tuple_len	
dead_tuple_percent	float8	pgstattuple().dead_tuple_percent	
free_space	int8	pgstattuple().free_space	
free_percent	float8	pgstattuple().free_percent	

pgperf.snapshot_pgstatindex Table

This table stores snapshots of the result of the `pgstatindex()` function which gets the index fragmentation statistics. This table is available only when the `pgstattuple` module has been installed.

Column Name	Type	Source	Note
sid	integer	Snapshot ID	
schemaname	name	pg_stat_user_indexes.schemaname	
relname	name	pg_stat_user_indexes.relname	
indexrelname	name	pg_stat_user_indexes.indexrelname	
version	int4	pgstatindex().version	
tree_level	int4	pgstatindex().tree_level	
index_size	int8	pgstatindex().index_size	
root_block_no	int8	pgstatindex().root_block_no	
internal_pages	int8	pgstatindex().internal_pages	
leaf_pages	int8	pgstatindex().leaf_pages	
empty_pages	int8	pgstatindex().empty_pages	
deleted_pages	int8	pgstatindex().deleted_pages	
avg_leaf_density	float8	pgstatindex().avg_leaf_density	
leaf_fragmentation	float8	pgstatindex().leaf_fragmentation	

14.1 Summary

Displays status information about the table

14.2 Usage

```
pt-table-usage [option...]
```

14.3 Options

```
-h, --host=HOSTNAME  
-p, --port=PORT  
-U, --username=USERNAME  
-d, --dbname=DBNAME  
-o, --owner=STRING  
-n, --schema=STRING  
-t, --table=STRING  
--help
```

-h, --host Specifies the connecting PostgreSQL database server name or IP address. If not specified, to set the value of PGHOST. localhost will be used if no value has been defined in PGHOST.

-p, --port Specifies the port number of the connecting PostgreSQL database. If not specified, to set the value of PGPORT. 5432 will be used if no value has been defined in PGPORT.

-U, --username Specifies the user name of the PostgreSQL database. If not specified, to set the value of PGUSER. The value of USER will be used if no value has been defined in PGUSER.

-d, --dbname Specifies the connecting database name. If not specified, to set the value of PGDATABASE. It connect to the database same as the database username if no value has been defined in PGDATABASE.

-o, --owner Displays only table that is the owner user that matches the specified name. Also accepts regular expression when starting and ending with slashes. (cf. /REGEXP/)

-n, --schema Displays only table that is schema that matches the specified name. Also accepts regular expression when starting and ending with slashes. (cf. /REGEXP/)

-t, --table Displays only table that matches the specified name. Also accepts regular expression when starting and ending with slashes. (cf. /REGEXP/)

-d (or --dbname), -o (or --owner), -n (or --schema), -t (or --table), if specified the options at the same time, only table that matches the conditions.

14.4 Output Items

OID	Table object ID
OWNER	Owner name of table
SCHEMA	Schema name that exists in the table
TABLE	Table name
BLKS	Number of the table block (every 8kb)
SCAN	Sequential scan number of executions
T_READ	Number of tuple by sequential scan
T_INS	Number of inserted tuple
T_UPD	Number of updated tuple(include HOT UPDATE)
T_DEL	Number of delated tuple
B_READ	Number of read table block from disk
B_HIT	Number of read table page by shared buffer
VACUUMED	Date and time that run VACUUM on the last (VACUUM command or automatic VACUUM)
ANALYZED	Date and time that run ANALYZE on the last (ANALYZE command or automatic ANALYZE)
TABLESPACE	The tablespace name where the table is placed.

14.5 Examples

It connects to the PostgreSQL instance running on the port 5432 of localhost, and to display all of the table usage of dbt3 database.

```
$ pt-table-usage -d dbt3
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|  OID  | OWNER | SCHEMA | TABLE | BLKS | SCAN | T_READ | T_INS | T_UPD | T_DEL | B_READ | B_HIT | VACUUMED | ANALYZED | TABLESPACE |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1273410 | snaga | public | customer | 3531 | 5 | 750000 | 150000 | 0 | 0 | 6499 | 29943 | 2015-03-08 18:31:41 | ssdspc1 |
| 1273416 | snaga | public | lineitem | 106583 | 12 | 66656465 | 6001215 | 0 | 0 | 240547 | 1340871 | 2015-03-08 18:31:42 | ssdspc1 |
| 1273419 | snaga | public | nation | 1 | 4 | 100 | 25 | 0 | 0 | 1 | 5 | 2015-03-08 18:31:42 | ssdspc1 |
| 1273413 | snaga | public | orders | 25326 | 5 | 7500000 | 1500000 | 0 | 0 | 48612 | 208386 | 2015-03-08 18:31:41 | ssdspc1 (continues on next page)
```


(continued from previous page)

```

| 1273404 | snaga | public | part      | 4064 | 3 | 600000 | 200000 | 0 |  |
↪ 0 | 6082 | 26558 |          | 2015-03-08 18:31:40 | ssdspc1 |
| 1273407 | snaga | public | partsupp | 17087 | 5 | 4000000 | 800000 | 0 |  |
↪ 0 | 32200 | 148518 |          | 2015-03-08 18:31:41 | ssdspc1 |
| 1273422 | snaga | public | region   | 1 | 3 | 15 | 5 | 0 |  |
↪ 0 | 1 | 4 |          | 2015-03-08 18:31:42 | ssdspc1 |
| 1273401 | snaga | public | supplier | 218 | 4 | 40000 | 10000 | 0 |  |
↪ 0 | 220 | 1802 |          | 2015-03-08 18:31:40 | ssdspc1 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
↪ +-----+-----+-----+-----+-----+-----+-----+-----+-----+
$

```

It connects to the PostgreSQL instance running on the port 5432 of localhost, and to display the table usage of the tables which start with pa in the dbt3 database.

```

$ pt-table-usage -d dbt3 -t /^pa/
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
↪ +-----+-----+-----+-----+-----+-----+-----+-----+-----+
|  OID  | OWNER | SCHEMA | TABLE  | BLKS | SCAN | T_READ | T_INS | T_UPD | T_
↪DEL | B_READ | B_HIT | VACUUMED |      | ANALYZED | TABLESPACE |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
↪ +-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1273404 | snaga | public | part      | 4064 | 3 | 600000 | 200000 | 0 |  |
↪0 | 6082 | 26558 |          | 2015-03-08 18:31:40 | ssdspc1 |
| 1273407 | snaga | public | partsupp | 17087 | 5 | 4000000 | 800000 | 0 |  |
↪0 | 32200 | 148518 |          | 2015-03-08 18:31:41 | ssdspc1 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
↪ +-----+-----+-----+-----+-----+-----+-----+-----+-----+
$

```


15.1 Summary

Displays the usage of table space for each databases.

For each table space, it will be able to check which database is using how much the capacity.

15.2 Usage

```
pt-tablespace-usage [option...]
```

15.3 Options

```
-h, --host=HOSTNAME  
-p, --port=PORT  
-U, --username=USERNAME  
-d, --dbname=DBNAME  
--help
```

-h, --host Specifies the connecting PostgreSQL database server name or its IP address. If not specified, to set the value of PGHOST. localhost will be used if no value has been defined in PGHOST.

-p, --port Specifies the port number of the connecting PostgreSQL database. If not specified, to set the value of PGPORT. 5432 will be used if no value has been defined in PGPORT.

-U, --username Specifies the user name of the PostgreSQL database. If not specified, to set the value of PGUSER. The value of USER will be used if no value has been defined in PGUSER.

-d, --dbname Specifies the connecting database name. If not specified, to set the value of PGDATABASE. It connect to the database same as the database username if no value has been defined in PGDATABASE.

15.4 Output Items

TABLESPACE	Tablespace name
DBNAME	Database name (The blank in case of pg_global table space)
SIZE (MB)	The capacity that is using the database object on the table space.(MB)

15.5 Examples

It connects to the `postgres` database of PostgreSQL instance running on the port 5432 of localhost, and to display the capacity that they are using for each database in each tablespace.

```
$ pt-tablespace-usage -d postgres
+-----+-----+-----+
| TABLESPACE | DBNAME | SIZE (MB) |
+-----+-----+-----+
| pg_default  | postgres |      8 |
| pg_default  | template1 |      6 |
| pg_default  | testdb   |      8 |
| pg_global   |          |      1 |
| spcl        | postgres |     16 |
+-----+-----+-----+
```

16.1 Summary

Verifies the checksum of the specified PostgreSQL file

Make sure to have created a database cluster that enable the checksum. (-k option of `initdb` command)

Version PostgreSQL 9.3 or higher

The script invoke `verifychecksum` command inside. If it run on other OS than Hat Enterprise Linux 6/CentOS 6, you must separately build to `src/verifychecksum.c` to be located in the following `bin` directory.

16.2 Usage

```
pt-verify-checksum [option...] [segment file | directory]
```

If specified the file name as an argument, verify the checksum.

If specified a directory name as an argument, of the files in that directory, to verify the checksum as the subject of the following files.

- The files found in the `global` and `base` directories, and
- The file name is composed of a number (ex. `1234`, `'1234.1'`), and it contained that `_vm` or `_fsm` are appended at the end of files.

If errors were found in one or more files in the checksum verification, return exit code 1. If any other error occurred, return exit code 2. If there are no problems in any files, return 0.

16.3 Options

```
-r, --recursive
-v, --verbose
--help
```

`-r, --recursive` If options specified directory, recursively scan the sub-directory below, to verify the checksum of the applicable file.

`-v, --verbose` During the checksum verification, it will show output the more messages.

16.4 Output Items

blkno	Block number that found the checksum error
expected	Checksum calculated from the data of the block
found	Checksum which had been recorded in the page header
Verified N files	Number of files that the checksum verification
N files corrupted	Number of files that found the checksum error

16.5 Examples

Verifies the checksum of single file.

```
$ pt-verify-checksum /var/lib/pgsql/9.4/data/base/16386/16399
[2015-03-28 15:50:03] INFO: Verified 1 files. 0 files corrupted.
$
```

Verifies the checksum of all of files in the database.

```
$ pt-verify-checksum /var/lib/pgsql/9.4/data/base/16386
[2015-03-28 15:51:00] INFO: Verified 311 files. 0 files corrupted.
$
```

It recursively searched in the database cluster, and to verify the checksum of all of files.

```
$ pt-verify-checksum -r /var/lib/pgsql/9.4/data
[2015-03-28 15:55:16] INFO: /var/lib/pgsql/9.4/data/base/12144/11905: blkno 7,
↪ expected 2cf, found da97
[2015-03-28 15:55:16] INFO: 1 blocks corrupted in /var/lib/pgsql/9.4/data/base/12144/
↪ 11905.
[2015-03-28 15:55:16] INFO: Verified 1046 files. 1 files corrupted.
$
```

17.1 Summary

Displays transaction statistical information of multiple node. It is possible to display each specified interval continuously.

17.2 Usage

```
pt-xact-stat [option...] [delay [count]]
```

17.3 Options

```
-h, --host=HOSTNAME  
-p, --port=PORT  
-H, --host-list=HOSTLIST  
-U, --username=USERNAME  
-d, --dbname=DBNAME  
-H, --host-list=HOSTNAME:PORT, HOSTNAME:PORT [, ...]
```

-h, --host Specifies the connecting PostgreSQL database server name or its IP address. If not specified, to set the value of PGHOST. localhost will be used if no value has been defined in PGHOST.

-p, --port Specifies the port number of the connecting PostgreSQL database. If not specified, to set the value of PGPORT. 5432 will be used if no value has been defined in PGPORT.

-H, --host-list If it is connected to multiple database servers, it specifies database server name or several combination IP address or port number. The form is 192.168.1.101:5432,192.168.1.102:5433 . A set of server and port number is linked with a colon. A set of several server is linked with a comma. This port number is optional and will be set for the default port number.

-U, --username Specifies the user name of the PostgreSQL database. If not specified, to set the value of PGUSER. The value of USER will be used if no value has been defined in PGUSER.

-d, --dbname Specifies the connecting database name. If not specified, to set the value of PGDATABASE. It connect to the database same as the database username if no value has been defined in PGDATABASE.

17.4 Output Items

HOST	Host name of PostgreSQL server
PORT	Port number of PostgreSQL server
DBNAME	Database name
CONN	Number of sessions that are connected to the database
COMMITTS	Total number of transactions that are committed
ROLLBACKS	Total number of transactions that are rollback
B_READ	Number of blocks in the table that read from disk
B_HIT	Number of pages that read from shared buffer

17.5 Examples

It connects to the two PostgreSQL instance running on the port 5432 and port 5433 of the localhost, and to display the statistical information of transaction for each instance every 5 seconds twice.

```
$ pt-xact-stat --host-list 127.0.0.1:5432,127.0.0.1:5433,127.0.0.1:5434 -d postgres 5_
↪2
Sat Mar 28 20:47:50 JST 2015
+-----+-----+-----+-----+-----+-----+-----+-----+
|  HOST   | PORT | DBNAME | CONN | COMMITTS | ROLLBACKS | B_READ | B_HIT |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 127.0.0.1 | 5432 | postgres | 1 | 137 | 1 | 104 | 10273 |
| 127.0.0.1 | 5433 | postgres | 1 | 8 | 0 | 104 | 1350 |
| 127.0.0.1 | 5434 | postgres | 1 | 76 | 0 | 104 | 7708 |
+-----+-----+-----+-----+-----+-----+-----+-----+

Sat Mar 28 20:47:55 JST 2015
+-----+-----+-----+-----+-----+-----+-----+-----+
|  HOST   | PORT | DBNAME | CONN | COMMITTS | ROLLBACKS | B_READ | B_HIT |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 127.0.0.1 | 5432 | postgres | 1 | 139 | 1 | 104 | 10460 |
| 127.0.0.1 | 5433 | postgres | 1 | 10 | 0 | 104 | 1537 |
| 127.0.0.1 | 5434 | postgres | 1 | 78 | 0 | 104 | 7895 |
+-----+-----+-----+-----+-----+-----+-----+-----+

$
```

Connects to the three PostgreSQL instance running on the port 5432, port 5433 and port 5434 of the localhost, and to display once the statistical information of transaction for each instance, and exit.

```
$ pt-xact-stat --host-list 127.0.0.1:5432,127.0.0.1:5433,127.0.0.1:5434 -d postgres
Sat Mar 28 21:05:48 JST 2015
+-----+-----+-----+-----+-----+-----+-----+-----+
|  HOST   | PORT | DBNAME | CONN | COMMITTS | ROLLBACKS | B_READ | B_HIT |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 127.0.0.1 | 5432 | postgres | 1 | 139 | 1 | 104 | 10460 |
| 127.0.0.1 | 5433 | postgres | 1 | 10 | 0 | 104 | 1537 |
| 127.0.0.1 | 5434 | postgres | 1 | 78 | 0 | 104 | 7895 |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

(continues on next page)

(continued from previous page)

	127.0.0.1		5432		postgres		1		12		0		104		1400	
	127.0.0.1		5433		postgres		1		4		0		104		976	
	127.0.0.1		5434		postgres		1		4		0		104		976	
+-----+-----+-----+-----+-----+-----+-----+-----+																
\$																

18.1 Version 0.2.1

- Add support for Red Hat Enterprise Linux 7, CentOS 7 and Ubuntu 14.04 LTS with Python 2.7.
- Fix to correct help messages. (pt-index-usage, pt-set-tablespace, pt-table-usage)

18.2 Version 0.2.0

- First Release Version

CHAPTER 19

Indices and tables

- genindex
- modindex
- search