

Upgrade PostgreSQL on Ubuntu

In this example we're upgrade psql from version 10 to version 11.

Install the new version

```
root@srv01:~# apt install postgresql-11
```

List psql clusters

```
root@srv01:~# pg_lsclusters
```

Ver	Cluster	Port	Status	Owner	Data directory	Log file
10	main	5432	down	postgres	/var/lib/postgresql/10/main	/var/log/postgresql/postgresql-10-main.log
11	main	5433	online	postgres	/var/lib/postgresql/11/main	/var/log/postgresql/postgresql-11-main.log

Backup psql

```
root@srv01:~# su - postgres
postgres@srv01:~$ pg_dumpall > psql_dump.sql
```

Stop the old cluster

```
root@srv01:~# pg_ctlcluster 10 main stop
```

Drop the new created cluster

```
root@srv01:~# pg_dropcluster --stop 11 main
```

Upgrade the old Cluster

```
root@srv01:~# pg_upgradecluster -m upgrade 10 main
Disabling connections to the old cluster during upgrade...
Restarting old cluster with restricted connections...
Stopping old cluster...
Creating new PostgreSQL cluster 11/main ...
/usr/lib/postgresql/11/bin/initdb -D /var/lib/postgresql/11/main --auth-local peer --auth-host md5 --encoding UTF8
--lc-collate en_US.UTF-8 --lc-ctype en_US.UTF-8
The files belonging to this database system will be owned by user "postgres".
This user must also own the server process.
```

The database cluster will be initialized with locale "en_US.UTF-8".
The default text search configuration will be set to "english".

Data page checksums are disabled.

```
fixing permissions on existing directory /var/lib/postgresql/11/main ... ok
creating subdirectories ... ok
selecting default max_connections ... 100
selecting default shared_buffers ... 128MB
selecting dynamic shared memory implementation ... posix
creating configuration files ... ok
running bootstrap script ... ok
performing post-bootstrap initialization ... ok
syncing data to disk ... ok
```

Success. You can now start the database server using:

```
/usr/lib/postgresql/11/bin/pg_ctl -D /var/lib/postgresql/11/main -l logfile start
```

Ver	Cluster	Port	Status	Owner	Data directory	Log file
11	main	5433	down	postgres	/var/lib/postgresql/11/main	/var/log/postgresql/postgresql-11-main.log

```
/usr/lib/postgresql/11/bin/pg_upgrade -b /usr/lib/postgresql/10/bin -B /usr/lib/postgresql/11/bin -p 5432 -P 5433 -
d /etc/postgresql/10/main -D /etc/postgresql/11/main
```

Finding the real data directory for the source cluster ok

Finding the real data directory for the target cluster ok

Performing Consistency Checks

Checking cluster versions ok

Checking database user is the install user ok

Checking database connection settings ok

Checking for prepared transactions ok

Checking for reg* data types in user tables ok

Checking for contrib/isn with bigint-passing mismatch ok

Creating dump of global objects ok

Creating dump of database schemas

ok

Checking for presence of required libraries ok

Checking database user is the install user ok

Checking for prepared transactions ok

If pg_upgrade fails after this point, you must re-initdb the
new cluster before continuing.

Performing Upgrade

Analyzing all rows in the new cluster ok

Freezing all rows in the new cluster ok

Deleting files from new pg_xact ok

Copying old pg_xact to new server ok

Setting next transaction ID and epoch for new cluster ok

Deleting files from new pg_multixact/offsets ok

Copying old pg_multixact/offsets to new server ok

Deleting files from new pg_multixact/members ok

Copying old pg_multixact/members to new server ok

Setting next multixact ID and offset for new cluster ok

Resetting WAL archives ok

Setting frozenxid and minmxid counters in new cluster ok

Restoring global objects in the new cluster ok

Restoring database schemas in the new cluster

ok

Copying user relation files

ok

Setting next OID for new cluster ok

```
Sync data directory to disk          ok
Creating script to analyze new cluster      ok
Creating script to delete old cluster      ok
```

Upgrade Complete

Optimizer statistics are not transferred by pg_upgrade so,
once you start the new server, consider running:

```
./analyze_new_cluster.sh
```

Running this script will delete the old cluster's data files:

```
./delete_old_cluster.sh
```

pg_upgrade output scripts are in /var/log/postgresql/pg_upgradecluster-10-11-main.5n35

Re-enabling connections to the old cluster...

Copying old configuration files...

Copying old start.conf...

Copying old pg_ctl.conf...

Disabling automatic startup of old cluster...

Configuring old cluster to use a different port (5433)...

Success. Please check that the upgraded cluster works. If it does,
you can remove the old cluster with

```
pg_dropcluster 10 main
```

Ver	Cluster	Port	Status	Owner	Data directory	Log file
10	main	5433	down	postgres	/var/lib/postgresql/10/main	/var/log/postgresql/postgresql-10-main.log

Ver	Cluster	Port	Status	Owner	Data directory	Log file
11	main	5432	down	postgres	/var/lib/postgresql/11/main	/var/log/postgresql/postgresql-11-main.log

Start new cluster

```
root@srv01:~# pg_ctlcluster 11 main start
```

Post Checks

Check is the new cluster is working properly, when ready delete the old cluster

```
root@srv01:~# apt-get autoremove --purge postgresql-10
Reading package lists... Done
Building dependency tree
```

Reading state information... Done

The following packages will be REMOVED:

postgresql-10*

0 upgraded, 0 newly installed, 1 to remove and 0 not upgraded.

After this operation, 23.5 MB disk space will be freed.

Do you want to continue? [Y/n]

(Reading database ... 39243 files and directories currently installed.)

Removing postgresql-10 (10.6-1.pgdg16.04+1) ...

Purging configuration files for postgresql-10 (10.6-1.pgdg16.04+1) ...

Dropping cluster main...

Processing triggers for postgresql-common (197.pgdg16.04+1) ...

Building PostgreSQL dictionaries from installed myspell/hunspell packages...

Removing obsolete dictionary files:

Revision #3

Created 12 February 2021 07:40:18

Updated 17 May 2023 11:57:16 by Peter Baumann