

# EXploitation des BDD.

TABLESPACE , INDEX, SCHEMA, UTILISATEURS, DROITS

# Processus Server postgresql

## ■ Gestion du lancement du serveur Postgresql

```

service postgresql status
service postgresql stop
service postgresql start
service postgresql reload
    
```

```

dany@dany-HP-ZBook-15-G2:~$ service postgresql status
● postgresql.service - PostgreSQL RDBMS
   Loaded: loaded (/lib/systemd/system/postgresql.service; enabled; vendor preset: enabled)
   Active: active (exited) since Mon 2022-04-04 09:34:25 CEST; 18min ago
   Process: 1765 ExecStart=/bin/true (code=exited, status=0/SUCCESS)
   Main PID: 1765 (code=exited, status=0/SUCCESS)

avril 04 09:34:25 dany-HP-ZBook-15-G2 systemd[1]: Starting PostgreSQL RDBMS...
avril 04 09:34:25 dany-HP-ZBook-15-G2 systemd[1]: Finished PostgreSQL RDBMS.
dany@dany-HP-ZBook-15-G2:~$
    
```

# Répertoires de postgresql (Linux)

## ■ Répertoire de la configuration du SGDBR

```
dany@dany-HP-ZBook-15-G2:~$ cd /etc/postgresql
dany@dany-HP-ZBook-15-G2:/etc/postgresql$ ls
10 12 9.5
dany@dany-HP-ZBook-15-G2:/etc/postgresql$ cd 12
dany@dany-HP-ZBook-15-G2:/etc/postgresql/12$ ls
main
dany@dany-HP-ZBook-15-G2:/etc/postgresql/12$ cd main
dany@dany-HP-ZBook-15-G2:/etc/postgresql/12/main$ ls
conf.d environment pg_ctl.conf pg_hba.conf pg_ident.conf postgresql.conf start.conf
dany@dany-HP-ZBook-15-G2:/etc/postgresql/12/main$ less postgresql.conf
dany@dany-HP-ZBook-15-G2:/etc/postgresql/12/main$ less pg_hba.conf
pg_hba.conf: Permission non accordée
dany@dany-HP-ZBook-15-G2:/etc/postgresql/12/main$ sudo less pg_hba.conf
[sudo] Mot de passe de dany :
dany@dany-HP-ZBook-15-G2:/etc/postgresql/12/main$ █
```

# Répertoires de postgresql (Linux)

## ■ Répertoire des Datas

```
CALL
dany@dany-HP-ZBook-15-G2:~$ sudo su postgres
postgres@dany-HP-ZBook-15-G2:/home/dany$ cd /var/lib/postgresql/12/main/
postgres@dany-HP-ZBook-15-G2:~/12/main$ ls
base          pg_multixact  pg_stat       PG_VERSION    postmaster.pid
global        pg_notify     pg_stat_tmp   pg_wal
pg_commit_ts  pg_replslot   pg_subtrans   pg_xact
pg_dynshmem   pg_serial     pg_tblspc     postgresql.auto.conf
pg_logical    pg_snapshots  pg_twophase    postmaster.opts
postgres@dany-HP-ZBook-15-G2:~/12/main$
```

# Répertoires de postgresql (Linux)

## ■ Répertoire des BIN (programmes)

```
dany@dany-HP-ZBook-15-G2:~$
dany@dany-HP-ZBook-15-G2:~$ sudo su postgres
postgres@dany-HP-ZBook-15-G2:/home/dany$ cd /usr/lib/postgresql/
postgres@dany-HP-ZBook-15-G2:/usr/lib/postgresql$ ls
10 12 9.5
postgres@dany-HP-ZBook-15-G2:/usr/lib/postgresql$ cd 12
postgres@dany-HP-ZBook-15-G2:/usr/lib/postgresql/12$ ls
bin lib
postgres@dany-HP-ZBook-15-G2:/usr/lib/postgresql/12$ cd bin
postgres@dany-HP-ZBook-15-G2:/usr/lib/postgresql/12/bin$ ls
clusterdb      pg_archivecleanup  pg_dump          pg_rewind        postmaster
createdb       pg_basebackup      pg_dumpall       pg_standby       psql
createuser     pgbench            pg_isready       pg_test_fsync    reindexdb
dropdb         pg_checksums       pg_receivewal    pg_test_timing   vacuumdb
dropuser       pg_config          pg_recvlogical   pg_upgrade       vacuumlo
initdb         pg_controldata     pg_resetwal      pg_waldump
oid2name       pg_ctl             pg_restore       postgres
postgres@dany-HP-ZBook-15-G2:/usr/lib/postgresql/12/bin$ cd psql
bash: cd: psql: N'est pas un dossier
postgres@dany-HP-ZBook-15-G2:/usr/lib/postgresql/12/bin$ psql
.psql (12.9 (Ubuntu 12.9-0ubuntu0.20.04.1), server 9.5.13)
Type "help" for help.

postgres=# .ls
```

## Binaires utiles (Linux)

- PG\_CTL (permet de démarrer ou d'arrêter une instance postgresql )
- PSQL permet d'avoir accès à un client qui permet de se connecter à un cluster postgresql en mode commande

# Binaires utiles (Linux)

- PG\_DUMP permet la sauvegarde d'une instance avec des spécifications ( plain text sql avec plusieurs niveaux de sauvegarde.
- PG\_DUMPALL sort un format binaire qui sauvegarde l'intégralité des données
- PG\_RESTORE Restaurer une BDD

# Les Wrappers (équivalent commande sql sans être connecté à la base)

- Createdb en ligne de commande shell
- Createuser
- Dropdb



# Les commandes de maintenance

- Reindexdb réindexer les index
- Vaccumdb = tâche de maintenance (ménage)

# Les commandes spécifiques avancées

- Pg\_controldata : vérifie l'état du serveur et les infos critiques

# TABLESPACE. Définition

- Allocation d'espace disque pour la BDD
- Stockage des tables dans cet espace
- Tout objet de la base est stocké dans un et un seul tablespace

# TABLESPACE. Utilité

- Disque dur presque saturé, création d'un tablespace dans un autre disque.
- Optimisation des performances. Création d'un tablespace pour les index sur un disque dur rapide

# TABLESPACE. Syntaxe

- CREATE TABLESPACE <nom>  
LOCATION ' chemin du fichier' ;

```
postgres=# CREATE TABLESPACE mondisque OWNER dany LOCATION '/home/dany/Documents/data';  
ERROR: could not set permissions on directory "/home/dany/Documents/data": Opération no  
n permise
```

- Attention aux droits !!

# TABLESPACE. Syntaxe

- Par défaut , les BDD sont stockées sur le TABLESPACE pg\_default

```
postgres=# SELECT spcname FROM pg_tablespace;  
spcname  
-----  
pg_default  
pg_global  
(2 rows)
```

# TABLESPACE. Syntaxe

- Afficher les BDD contenus dans un TABLESPACE

```
postgres=# select datname AS database , spcname AS tablespace FROM
postgres=# pg_database INNER JOIN pg_tablespace ON
postgres=# pg_tablespace.oid=pg_database.dattablespace;
 database | tablespace
-----+-----
 template1 | pg_default
 template0 | pg_default
 postgres  | pg_default
 dcapitainedb | pg_default
(4 rows)
```

# INDEX : WHAT IS AN INDEX ?

- Afficher le temps mis par chaque requête à l'aide de la commande \timing
- Un index améliore la performance des requêtes SQL
- Un index est comparable à index de livre
- Il accélère la recherche en évitant au SGBD de « parcourir ou scanner inutilement la table dans son intégrité ».



# INDEX : COMMENT LE CRÉER ?

■ CREATE INDEX nom\_index  
ON nom\_table (nom\_colonne);

OPTIONNEL

```
dcapitainedb=# SELECT numetu, nometu from etudiants;
Time: 3,293 ms
dcapitainedb=# CREATE INDEX index1 ON etudiants (numetu,nometu);
CREATE INDEX
Time: 65,330 ms
dcapitainedb=# SELECT numetu, nometu from etudiants;
Time: 0,350 ms
dcapitainedb=#
```

GAIN DE TEMPS SUR LA REQUÊTE

# INDEX : AFFICHER LES INDEX(S)

```
dcapitainedb=# \di
```

List of relations

Schema	Name	Type	Owner	Table
public	avoir_note_pkey	index	postgres	avoir_note
public	enseignants_pkey	index	postgres	enseignants
public	epreuves_pkey	index	postgres	epreuves
public	etudiants_pkey	index	postgres	etudiants
public	faire_cours_pkey	index	postgres	faire_cours
public	index1	index	postgres	etudiants
public	matieres_pkey	index	postgres	matieres
public	modules_pkey	index	postgres	modules

(8 rows)

```
dcapitainedb=# \di index1;
```

List of relations

Schema	Name	Type	Owner	Table
public	index1	index	postgres	etudiants

(1 row)

# INDEX : SUPPRIMER UN INDEX

- DROP INDEX nom\_index;

```
dcapitainedb=# DROP INDEX index1;  
DROP INDEX  
Time: 65,382 ms  
dcapitainedb=#
```

# INDEX : TYPE D'INDEX

- Il est possible de créer un index sur une ou plusieurs colonnes d'une table
- Possible de créer un index partiel de table

```
dcapitainedb=# CREATE INDEX index_partiel ON etudiants (numetu)
dcapitainedb=# WHERE numetu<10;
CREATE INDEX
Time: 24.400 ms
```

# SCHEMA : Principes

- Un schéma est un espace de travail nommé
- Une BDD peut contenir 0, 1 ou plusieurs schéma contenant des tables
- Le nom d'une table dans un schéma est alors appelé 'nomschema.nomtable'
- Il peut exister deux tables de même nom dans une BDD si elles sont dans un schéma différent.
- Un utilisateur peut accéder à tous les schémas s'il en a les droits !!

# SCHEMA : SYNTAXE

- CREATE SCHEMA nom\_schema ;
- Par défaut, toutes les tables sont automatiquement mis dans un schéma appelé « PUBLIC ». Toute nouvelle BDD contient par défaut ce schema

CREATE TABLE produits (....);  
= CREATE TABLE public.produits (....);

# SCHEMA : SYNTAXE

- CREATE SCHEMA nom\_schema ;
- Afficher la liste des schemas contenus dans la BDD : \dn ou \dn+

```
dcapitainedb=# \dn
List of schemas
Name      | Owner
-----+-----
public    | postgres
schema1   | postgres
(2 rows)

dcapitainedb=#
```

- Savoir dans quel schéma nous travaillons :  
SHOW search\_path;

```
dcapitainedb=# show search_path;
search_path
-----
"$user",public
(1 row)
```

# SCHEMA : SYNTAXE

- Travailler dans un autre schéma que public :  
SET search\_path TO monschema ;

```
Time: 0,328 ms
dcapitainedb=# \dn
List of schemas
Name      | Owner
-----+-----
public    | postgres
schema1   | postgres
(2 rows)

dcapitainedb=# SET search_path TO schema1;
SET
Time: 0,331 ms
dcapitainedb=# show search_path ;
search_path
-----
schema1
(1 row)
```



# SCHEMA : SUPPRIMER

- Pour supprimer un schéma et les objets qu'il contient :

**DROP SCHEMA schema1 CASCADE ;**

- Pour supprimer une table spécifique dans un schema :

**DROP TABLE mon\_schema.nomtable ;**

```
dcapitainedb=# create table schema1.table1 (numero integer);  
CREATE TABLE  
Time: 109,706 ms  
dcapitainedb=# DROP SCHEMA schema1 CASCADE ;  
NOTICE: drop cascades to table table1  
DROP SCHEMA
```

# SCHEMA : UTILITÉ

- Autoriser le travail de plusieurs utilisateurs
- Organiser les tables en groupes logiques  
OU protection des données sensibles
- Les applications peuvent être placés dans  
des schémas séparés
- Par défaut un utilisateur a accès au schéma  
public et aux schémas qu'il créés.

# PRIVILEGES :

- Règle de base :
- Un utilisateur ne peut effectuer une opération quelconque que s'il possède les droits.
- Donner des privilèges c'est accorder des droits sur les objets qui constituent la base

# UTILISATEURS :

- Un utilisateur est un compte par lequel on peut se connecter sur la BDD.
- `CREATE USER nom_utilisateur WITH .. Superuser, createdb, createrole, createuser, valid until 'date', IN ROLE nom_role`

# GESTION DES UTILISATEURS :

- Octroyer un privilège ou donner un rôle à un utilisateur
- SYNTAXE :

GRANT {<role> [,]}

TO {<user> | <role> | PUBLIC [,]}

[WITH GRANT OPTION]

# GESTION DES UTILISATEURS : UTILITÉ

- Répartir les responsabilités en terme de création, détention, utilisation des objets de la base.

- Celui qui crée l'objet en est le propriétaire

GRANT {<privilege> | ALL} [(col1, col2, ...)]

ON <objet> TO {<utilisateur> | <role> | PUBLIC}

[WITH GRANT OPTION]

Les privilèges sont : DELETE, INSERT, REFERENCES, SELECT, UPDATE

# GESTION DES UTILISATEURS : UTILITÉ

- Remarque : Les privilèges sont accordés à un utilisateur, à un rôle, à tout le monde (PUBLIC)

```

dcapitainedb=# CREATE ROLE superman WITH superuser;
CREATE ROLE
Time: 3,866 ms
dcapitainedb=# CREATE USER directeur with encrypted password 'supermotdepasse';
CREATE ROLE
Time: 4,801 ms
dcapitainedb=# GRANT superman TO directeur ;
GRANT ROLE
Time: 6,315 ms
dcapitainedb=# DROP user directeur,superman;
DROP ROLE
Time: 3,307 ms

```

# ENLEVER DES PRIVILEGES

## ■ SYNTAXE :

```
REVOKE {<privilege> | ALL} [(col1, col2,...)]  
ON <objet> FROM {<utilisateur> | <role> |  
PUBLIC}
```



# ENLEVER DES PRIVILEGES

```

dca pitainedb=# CREATE USER superman with encrypted password 'superpassword';
CREATE ROLE
Time: 4,183 ms
dca pitainedb=# GRANT ALL PRIVILEGES ON SCHEMA schema1 TO superman;
ERROR:  schema "schema1" does not exist
Time: 0,579 ms
dca pitainedb=# CREATE SCHEMA schema1;
CREATE SCHEMA
Time: 4,577 ms
dca pitainedb=# GRANT ALL PRIVILEGES ON SCHEMA schema1 TO superman;
GRANT
Time: 3,731 ms
dca pitainedb=# REVOKE ALL ON SCHEMA schema1 FROM superman;
REVOKE
Time: 5,208 ms

```