



**PostgreSQL,
quand ce n'est pas votre métier**

Table des matières

1	Licence.....	3
2	Auteur.....	3
3	« Mais je n'y connais rien ! ».....	4
4	Au menu.....	4
5	PostgreSQL.....	4
5.1	Un peu d'histoire.....	5
5.2	Versions.....	5
5.3	Fonctionnalités internes.....	6
5.4	Fonctionnalités utilisateurs.....	6
5.5	Fonctionnalités administrateurs.....	7
5.6	Fonctionnalités développeurs.....	7
5.7	Sa communauté.....	8
5.8	Sponsors.....	8
6	Installation.....	9
6.1	Au niveau matériel.....	9
6.2	Au niveau des disques.....	9
6.3	Au niveau du système d'exploitation.....	10
6.4	Au niveau du système de fichiers.....	10
6.5	Installation de PostgreSQL.....	10
6.6	Paquets à jour ?.....	11
6.7	Exemple.....	11
7	Configuration du système.....	11
8	Configuration de PostgreSQL.....	12
8.1	Traces.....	12
8.2	Mémoire.....	12
8.3	CHECKPOINT.....	12
8.4	Planificateur.....	13
8.5	Accès.....	13
8.6	Petite aparté sur fsync.....	13
8.7	Et c'est tout !.....	13
9	Maintenance.....	14
9.1	VACUUM.....	14
9.2	ANALYZE.....	14
9.3	REINDEX.....	15
9.4	Automatisation.....	15
9.5	Processus autovacuum.....	15
10	Sauvegarde.....	16
10.1	Sauvegardes logiques.....	16
10.2	Sauvegardes physiques.....	16
11	Supervision.....	17
11.1	Statistiques.....	17
11.2	Traces.....	17
12	Ce qu'il faut bien comprendre.....	18

1 Licence



- Creative Common BY-NC-SA
- You are free
 - to Share
 - to Remix
- Under the following conditions
 - Attribution
 - Noncommercial
 - Share Alike

2 Auteur



- Guillaume Lelarge
- Travail
 - Directeur technique de Dalibo
 - email: guillaume.lelarge@dalibo.com
- Communauté
 - contributeur majeur du projet
 - vice-trésorier de PostgreSQL Europe
 - pgAdmin, la traduction de la documentation, l'organisation de pgconf.eu
 - email: guillaume@lelarge.info
 - twitter: [@g_lelarge](https://twitter.com/g_lelarge)

3 « Mais je n'y connais rien ! »



- Le monde des bases de données est souvent intimidant
- Et PostgreSQL, avec son côté strict, encore plus
- Et pourtant, PostgreSQL est simple
 - simple à installer
 - simple à configurer
 - simple à maintenir

4 Au menu



- PostgreSQL
- Installation
- Configuration
- Maintenance
- Sauvegarde
- Supervision

5 PostgreSQL



- Historique rapide
- Versions
- Fonctionnalités
- Communauté
- Sponsors

5.1 Un peu d'histoire



- Années 1970 : Ingres est développé à Berkeley
- 1985 : Postgres succède à Ingres
- 1995 : Ajout du langage SQL
- 1996 : Postgres devient PostgreSQL
- 1996 : Création du PostgreSQL Global Development Group

5.2 Versions



- 2 types de versions
 - mineure ou corrective
 - majeure
- 5 versions stables
 - 8.3, 8.4, 9.0, 9.1, 9.2
 - ne pas utiliser les versions précédentes !
- 1 version en cours de développement
 - la future 9.3

5.3 Fonctionnalités internes



- ACID
 - PostgreSQL complètement transactionnel
 - respecte ACID entièrement
- MVCC
 - utilisé par la majorité des SGBD
 - permet de lire et écrire sans se bloquer
 - permet la sauvegarde logique
- Journaux de transactions
 - toute modification y est enregistrée
 - permet de retrouver les données après un crash
 - permet la sauvegarde physique moteur allumé
 - permet la réplication

5.4 Fonctionnalités utilisateurs



- Standard SQL
 - objets: tables, vues, règles, séquences, triggers
 - opérations: jointures, sous-requêtes, requêtes CTE, requêtes Window, etc.
- Standard SQL/MED
 - tables distantes et ses connecteurs
- Va au-delà
 - différents types et algorithmes d'index
 - contrainte d'exclusion
 - nombreux types supplémentaires (inet, géométrique, range)

5.5 Fonctionnalités administrateurs



- Configuration
 - globale
 - par base, utilisateur, ou couple base/utilisateur
- Sécurité
 - en accès aux bases
 - en accès aux objets et aux données
- Réplication
 - asynchrone/synchrone
 - en cascade
- Plusieurs solutions
 - de sauvegarde
 - de réplication
 - de supervision

5.6 Fonctionnalités développeurs



- Grande extensibilité
 - types utilisateurs
 - procédures stockées
 - nombreux langages
 - nombreux types d'index

5.7 Sa communauté



- Core Team
 - 6 personnes
 - décident de la date de sortie des versions
 - rien de plus...
- Contributeurs majeurs
 - 38 personnes
 - dont les deux conférenciers ici présents
- Contributeurs
 - 40 personnes
 - internationaux
 - nombreuses différentes sociétés
- Contributeurs sur une version
 - plus d'une centaine

5.8 Sponsors



- Sponsors Platinum
 - 2ndQuadrant
 - Command Prompt
 - Dalibo
 - EnterpriseDB
 - Red Hat
- Mais aussi
 - Heroku, HP, NEC, NTT, Overblog, Redpill Linpro, Skype, etc.

6 Installation



- Matériel
- Système d'exploitation
- PostgreSQL

6.1 Au niveau matériel



- CPU
 - beaucoup d'utilisateurs \Rightarrow beaucoup de cores
 - grosses requêtes \Rightarrow grosse fréquence
 - gros cache L2/L3 de toute façon
- Mémoire
 - autant que vous pouvez

6.2 Au niveau des disques



- Disques
 - SATA/SAS OK
 - SSD à éviter
- Système RAID à tous les étages
 - RAID 1 pour le système
 - RAID 1 pour les journaux de transactions
 - RAID 10 pour les données
- Si cache en écriture
 - à activer si batterie présente
 - à désactiver sinon

6.3 Au niveau du système d'exploitation



- Disponible sur beaucoup de systèmes
 - Linux, BSD, Mac OS X, Windows, AIX, Solaris, etc.
- Utiliser de préférence Linux
 - peu importe la distribution
- Attention aux (mauvaises) performances avec Windows

6.4 Au niveau du système de fichiers



- Fonctionne sur tous les systèmes de fichiers
 - sauf FAT32
- De préférence, utiliser ext3/ext4/XFS
 - ext3/ext4 pour la stabilité
 - XFS pour les performances
- N'oubliez pas les options de montage
 - noatime par exemple

6.5 Installation de PostgreSQL



- Pas la peine de compiler PostgreSQL
- Utiliser les packages binaires déjà disponibles
 - installateur Windows d'EnterpriseDB
 - paquets Debian de votre distribution
 - paquets RPM de votre distribution

6.6 Paquets à jour ?



- Les paquets des distributions sont souvent en retard
- Pour Debian/Ubuntu
 - utiliser le PPA de Martin Pitt
- Pour Red Hat/Fedora/CentOS/etc
 - utiliser yum.postgresql.org
 - paquets conçus par Devrim Gunduz

6.7 Exemple



```
$ aptitude install postgresql-9.2
```

ou

```
$ yum install postgresql-9.2-server
```

7 Configuration du système



- Taille de la mémoire partagée
 - SHMALL et SHMMAX
- Contrôle de la mémoire partagée
 - dirty_ratio, dirty_background_ratio
 - dirty_bytes, dirty_background_bytes
- Sur-allocation de la mémoire
 - overcommit_memory et overcommit_ratio
- Comportement du swap
 - swappiness
- Utiliser /etc/sysctl.conf

8 Configuration de PostgreSQL



- Moteur (postgresql.conf)
 - traces
 - mémoire
 - CHECKPOINT
 - planificateur
- Accès (pg_hba.conf)

8.1 Traces



```
log_destination = 'stderr'  
log_directory = 'pg_log'  
logging_collector = on  
log_filename = 'postgres-%Y-%m-%d_%H%M%S'  
log_rotation_age = 1d  
log_rotation_size = 1GB  
log_min_duration_statement = 1s  
log_checkpoints = on  
log_connections = on  
log_disconnections = on  
log_lock_waits = on  
log_temp_files = 0
```

8.2 Mémoire



- shared_buffers
- work_mem
- maintenance_work_mem

8.3 CHECKPOINT



- checkpoint_segments
- checkpoint_timeout
- checkpoint_completion_target

8.4 Planificateur



- `effective_cache_size`
- `random_page_cost`

8.5 Accès



- Fichier `pg_hba.conf`
- Filtrage IP
- Authentification
 - interne (MD5)
 - externe (identd, LDAP, Kerberos, RADIUS, ...)
 - N'UTILISEZ PAS trust !
- Support natif de SSL

8.6 Petite aparté sur fsync



- Oui, le désactiver fait gagner en performance
- Mais vous ne le voulez pas
 - Vous préférez ne pas perdre vos données
 - NE TOUCHEZ PAS À fsync !
- Préférez-lui `synchronous_commit`

8.7 Et c'est tout !



- Il existe des tas d'autres paramètres
 - mais ils ne permettent pas de gagner autant que ceux-là
 - et peuvent poser soucis si mal configuré

9 Maintenance



- Trois opérations essentielles
 - VACUUM
 - ANALYZE
 - REINDEX
- Automatisable par cron
- ... et par autovacuum (pour les deux premiers)

9.1 VACUUM



- Lutter contre la fragmentation
- VACUUM
 - cartographie les espaces libres pour une prochaine réutilisation
 - utilisable en parallèle avec les autres opérations
- VACUUM FULL
 - défragmente la table
 - verrou exclusif

9.2 ANALYZE



- Met à jour les statistiques sur les données
- Utilisées par l'optimiseur de requêtes
- Échantillonnage
 - global : default_statistics_target
 - mais par colonne possible

9.3 REINDEX



- Lancer REINDEX régulièrement permet
 - de gagner de l'espace disque
 - d'améliorer les performances
 - de réparer un index corrompu
- VACUUM ne provoque pas de réindexation
- VACUUM FULL:
 - reindexe si PostgreSQL \geq 9.0
 - déséquilibre les index sinon

9.4 Automatisation



- Automatisation des tâches de maintenance
- Cron sous Unix
- Tâches planifiées sous Windows

9.5 Processus autovacuum



- Automatisation par cron
 - simple, voire simpliste
- Processus autovacuum
 - VACUUM/ANALYZE si nécessaire
 - nombreux paramètres
 - nécessite la récupération des statistiques d'activité

10 Sauvegarde



- 2 types de sauvegarde avec PostgreSQL
 - sauvegarde logique
 - sauvegarde physique
- Dans les 2 cas, solutions
 - à chaud
 - et cohérentes

10.1 Sauvegardes logiques



- pg_dump pour la sauvegarde
 - préférer le format personnalisé (ie, custom)
 - désactiver la décompression pour gagner du temps
- pg_dump -Fc -Z0 b1 > b1.dump
- pg_restore pour la restauration
 - restauration complète ou partielle
 - pg_restore -d b1 b1.dump

10.2 Sauvegardes physiques



- Sauvegarde PITR
 - archivage des journaux de transactions
 - sauvegarde des fichiers
- Plusieurs outils
 - pg_basebackup
 - omnipitr
 - pitrery
 - pgbarman

11 Supervision



- PostgreSQL fournit deux canaux d'information
 - les statistiques d'activité
 - les traces
- Mais pas d'historisation sur les statistiques
- Donc, nécessaire de mettre en place un outil externe

11.1 Statistiques



- Activer le collecteur de statistiques
- Vues statistiques
 - pg_stat_*
 - pg_statio_*
- Outils disponibles
 - pgstats
 - Zabbix, Nagios

11.2 Traces



- Bien configurer les traces
- Informations récupérables
 - requêtes avec leur durée d'exécution
 - requêtes en erreur
 - connexions, déconnexions
 - checkpoints
 - etc.
- Outils de rétro-analyse
 - tail_n_mail, logwatch
- Outils « live »
 - pgBagder, pgFouine, pgsi

12 Ce qu'il faut bien comprendre



- Difficile d'avoir une configuration réellement mauvaise
- Les problèmes de performances sont généralement dûs aux applications
- Ne pas être obsédé par le tuning
- Ne pas rester seul avec ses problèmes