



Développement d'une fonctionnalité pour PostgreSQL

Du besoin au commit...

Guillaume Lelarge
Consultant DBA

Parallélisation d'une requête

- Apparaît en version 9.6
- Exécution d'une requête sur plusieurs CPU
- S'améliore de version en version
- Certains manques sont comblés

Mais il reste quelques manques

- Quelques nœuds toujours pas parallélisés
- Les requêtes en écriture aux abonnés absents
- Pratiquement toutes les instructions DDL
- Et le fréquemment oublié : la supervision

La supervision aide...

- À savoir si la parallélisation est utilisée
- À mieux configurer le moteur et/ou les ressources

Quelques paramètres de configuration

- `max_worker_processes = 8`
- `max_parallel_workers = 8`
- `max_parallel_maintenance_workers = 2`
- `max_parallel_workers_per_gather = 2`
- `parallel_leader_participation = on`

Seule métrique récupérable (v13+)

```
SELECT
count(distinct leader_pid)
  AS parallel_queries,
count(*) FILTER (WHERE leader_pid IS NOT NULL)
  AS parallel_workers
FROM pg_stat_activity;
```

- Colonne leader_pid ajoutée par Julien Rouhaud
- Liens **commit** et **thread**

Mais ça ne répond pas à...

- Augmenter `max_parallel_workers` améliorera-t-il les performances du système ?
- Comment dois-je configurer `max_parallel_workers_per_gather` ?
- Le serveur a-t-il suffisamment/trop de CPU ?

Que faire pour améliorer les choses ?

- Sur un projet libre, réclamer une fonctionnalité a peu de chances d'aboutir
- Soit on le développe soi-même, soit on paie quelqu'un pour le faire

Dalibo et le développement

- Pas le cœur du métier des consultants de Dalibo
- Une connaissance générale du code pour les plus anciens « experts »
- Quelques patchs écrits pour des besoins spécifiques
 - par certaines personnes déjà suffisamment à l'aise avec le code, avec la communauté, et dotées d'une grande patience

Beaucoup d'intérêt des consultants

- Des demandes internes depuis trois ans environ
- Plusieurs réponses ont été apportées
 - Journées de hacking
 - Série « **Hacking PG** » sur le blog
 - Journée VIP Tech suite à la pgsession (Melanie, Bertrand, Julien)

Dalibo et sa R&D

- Chaque employé•e a la possibilité de consacrer 20% de son temps de travail à de la R&D
 - Soit sur le monde de PostgreSQL
 - Soit sur le monde des SCOP
- Très vaste...
 - écriture de patchs, traduction de documentation
 - veille technologique
 - préparation de conférences, organisation d'événements
 - etc



2 mars 2023

- Premiers travaux internes
- Benoit commence à travailler dessus pendant sa R&D
- et gratte des idées dans un wiki interne



Parallel Query Stats

What do we have

If we want a worker breakdown we have `pg_stat_activity.backend_type`.

- logical replication worker
- parallel worker, we can know if it's a maintenance tool or a user query by the query field of the leader ?
- background workers registered by extensions may have additional types (pick with a NOT IN the rest)

`pg_stat_activity.leader_pid` to link a bgw to it's leader.

is it enough?

- we see queries that have a bgworker not if they are //
 - has // worker = parallel query
 - doesnt have // worker = ???
- same problem as wait events: we know how many we we saw when the snapshot was taken not how many we we had.
 - don't know about what append between snapshot
 - don't know if a query spans several snapshots

Meetup Lille, Septembre 2023

- Conférence sur la parallélisation
- L'un des derniers slides indique :

DIFFICULTÉS

- Savoir si le parallélisme est utilisé
- Bien ou mal utilisé ?
- Bien ou mal configuré ?

Développement d'une fonctionnalité, du besoin au commit

5 avril 2023

- Premier thread sur la liste postgresql-hackers
- **Parallel Query Stats**
- Proposition de nouvelles statistiques d'activité
- Une réponse de Tomas Vondra, intéressé, et proposant quelques pistes de réflexion...
 - **... un an après !**

21 avril 2023

- Deuxième thread sur la liste pgsql-hackers
- **Logging parallel worker draught**
- Proposition d'une nouvelle trace pour chaque nœud parallélisé, si activation d'une option
- Une trentaine de réponses
 - ceux qui ne veulent pas d'un paramètre en plus
 - ceux qui n'aiment pas le nom du paramètre
 - ceux qui veulent un niveau de log différent
 - ceux qui ont peur que ça fasse trop de traces
 - ceux qui se demandent si ça ne devrait pas plutôt aller dans les stats

Janvier 2024

- Rencontre avec Melanie Plageman lors des pgsessions 16



Un petit coup d'arrêt

- Difficile d'avancer quand ce n'est pas son cœur de métier
- Surtout que Benoit y travaillait seul
- ... une semaine max par mois
- Donc ça traîne

Besoin interne

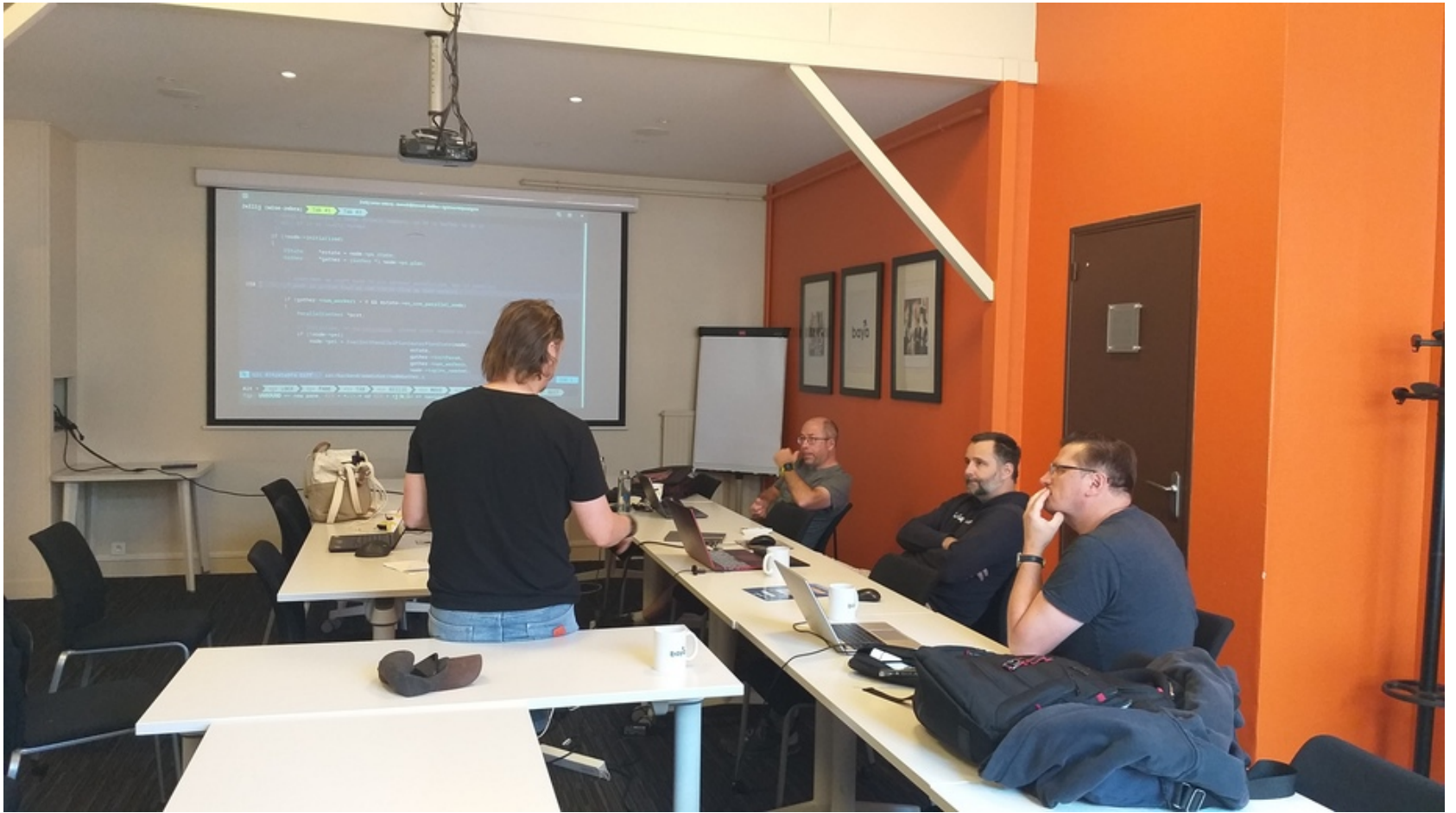
- Hacking PG est bien
- Mais pas forcément suffisant
 - 1 journée tous les un ou deux mois
 - Principalement sur des extensions
 - Pas vraiment sur le cœur de PostgreSQL
- Certains aimeraient passer plusieurs jours sur du développement pour le cœur de PostgreSQL

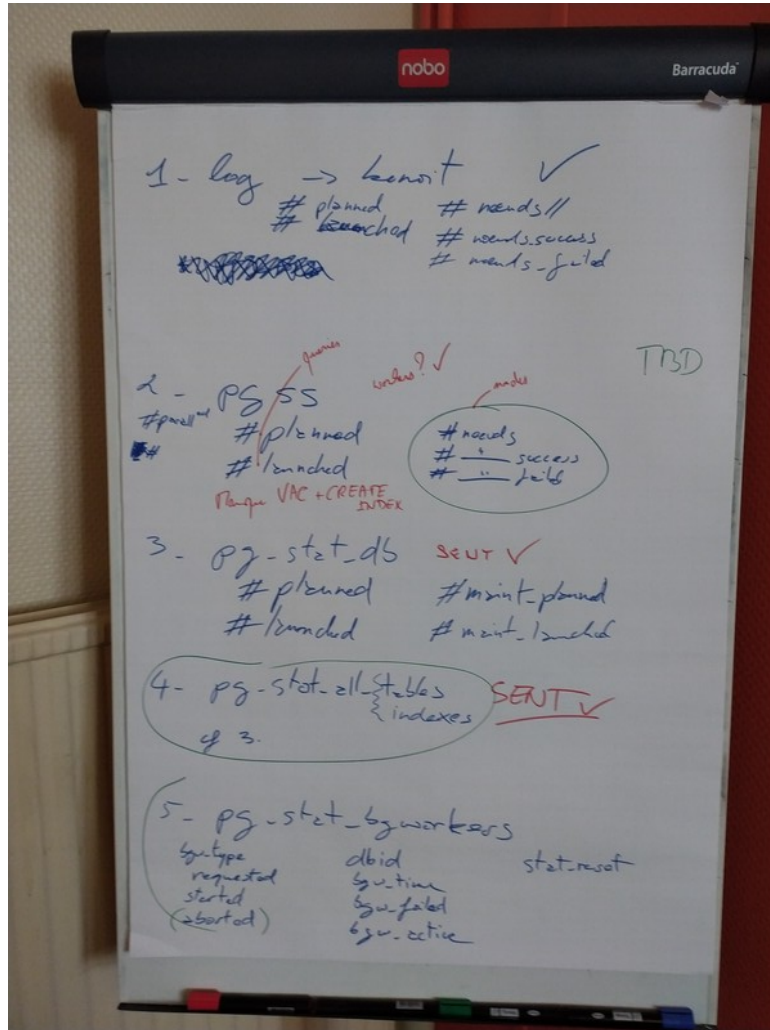
Dalibo et les coworks

- Télétravail exclusivement (ou presque)
- Mise en place de coworks mensuels à partir de 2022
- Une ville différente à chaque fois
- Pas forcément de thématique
 - Un moyen de faire du lien
 - Un moyen pour faciliter/accélérer certains projets

Cowork Dev PG

- Août 2024, sur 3 jours
- Nantes
- 4 personnes
- But : transformer en patchs concrets les idées de Benoît





Développement d'une fonctionnalité, du besoin au commit

Quatre morceaux

- Nouvelles traces sur la parallélisation
- Nouvelles statistiques pour `pg_stat_activity`
- Nouvelles statistiques pour `pg_stat_database`
- Nouvelles statistiques pour `pg_stat_all_tables/indexes`
- Et donc 4 patchs, et 4 threads

Première réaction

- Bertrand Drouvot, VIP tech de l'année dernière, qui est devenu contributeur majeur du projet en octobre
- Les patches intéressent, ils ne doivent pas être oubliés, ils sont enregistré dans un **commit fest**
- Michael Paquier, autre connaissance, s'intéresse à deux des patches, y voit des similarités
- Nouvelles versions pour certains patches
- Intégration d'un premier patch

Commit fest ?

- **Application web** permettant d'enregistrer les patchs à relire et intégrer
- Un commit fest tous les deux mois
 - Relecture des patchs
 - Conseils pour correction
 - Ou intégration dans le dépôt central

Patch #1

- Traces sur la parallélisation
- Thread
- Enregistrement commit fest
- Patch en version 5

Patch #2

- Nouvelles statistiques pour `pg_stat_database`
 - 4 nouvelles colonnes
- Thread
- Enregistrement `commit fest`
- Patch en version 6
- Commit `e7a9496de9`
 - Avec seulement deux colonnes sur les quatre

Patch #3

- Nouvelles statistiques pour pg_stat_all_tables/indexes
 - 4 colonnes pour les tables, 2 pour les index
- Thread
- Enregistrement commit fest
- Patch en version 5, refusé par Michael
 - Colonnes qui ne permettent pas d'améliorer la configuration
- État actuel, au mieux mort-vivant...

Patch #4

- Nouvelles statistiques pour `pg_stat_statements`
 - 7 nouvelles colonnes !
- Thread
- Enregistrement `commit fest`
- Commit `de3a2ea3b2` et Commit `cf54a2c002`
 - Avec seulement deux colonnes sur les quatre

Au final, la réussite

- 3 patches commités
- Le premier, infrastructure pour les deux autres
- Nouvelles colonnes pour `pg_stat_database` et `pg_stat_statements`
 - `parallel_workers_to_launch`
 - `parallel_workers_launched`

Mais aussi des échecs

- Plus de métriques n'est pas forcément mieux
 - Il faut réfléchir et savoir expliquer à quoi servent chaque ajout
- Dans le contexte de patches liés, s'assurer qu'on ne propose pas plusieurs fois la même infrastructure
 - Avoir une seule infrastructure
 - Même information, même nom de colonne, pour assurer l'homogénéité, pour faciliter l'utilisation

Ce qui reste à faire

- Patch sur les traces
 - J'ai toujours espoir :)
- Corriger les outils de supervision et d'audit
 - Pgcluu
 - check_pgactivity
 - PoWA
- Utiliser ces nouvelles informations
 - Et, le cas échéant, proposer les colonnes refusées, mais cette fois avec des arguments

Conseils pour des potentiels contributeurs

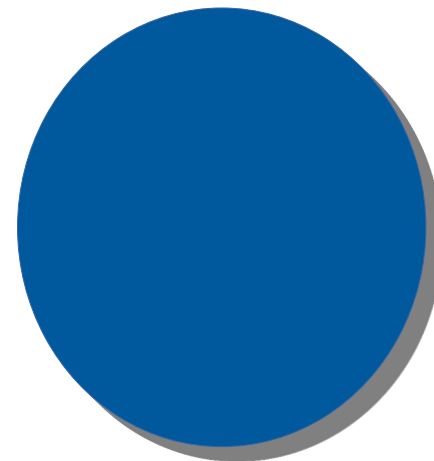
- Avoir de la patience
- S'attendre à être questionné sur tout et son contraire
- S'attendre à devoir tout justifier
- Envoyer dès le départ un mail complet expliquant le but recherché, les décisions prises et expliquant comment l'utiliser
 - [Posting Your Patch On postgresql-hackers](#), de Robert Haas
- Ne pas avoir d'ego est un plus confortable :)

Disclaimer final

Attention, ce qui est ajouté actuellement dans les sources de la future v18 peut être supprimé de la version finale !

Guillaume Lelarge

guillaume.lelarge@dalibo.com



À bientôt !