



Tout savoir sur max_connections

Guillaume Lelarge
Consultant DBA

Paramètre max_connections

- Certainement
 - le paramètre le plus connu
 - mais aussi le moins aimé :)

Définition de max_connections

- Nombre maximum de connexions à PostgreSQL
- Plus exactement
 - **nombre maximum de sessions à PostgreSQL à un instant t**
- Valeur par défaut 100
- Beaucoup souhaitent changer sa valeur
 - Mais ça a des conséquences

Impact RAM sur la mémoire partagée

- Plus `max_connections` est haut, plus la mémoire partagée est volumineuse
- Sur une version 17, configuration de base...

<code>max_connections</code>	Mémoire partagée
10	139 Mo
100	143 Mo
500	163 Mo
1000	184 Mo
10000	617 Mo
100000	4814 Mo

Impact RAM sur la mémoire privée

- Une session qui n'exécute rien ne consomme pas (beaucoup) de mémoire
- Une session qui exécute une requête consomme une mémoire privée
 - `work_mem`, `maintenance_work_mem`, etc.
- La mémoire privée totale peut être plus importante
 - augmenter `max_connections` peut forcer à diminuer `work_mem`

Impact CPU

- Une session = **Au moins** un processus
- Un processus s'exécute sur un CPU
- Donc nombre de sessions/processus = nombre de CPU ?
 - Non
 - **Un CPU va s'occuper de plusieurs processus** (donc de requêtes) en « **même temps** »

Impact CPU – nombre de processus

- Un `max_connections` à 2000 ne veut pas dire 2000 processus
- « 2000 processus présents » n'indique rien sur leur activité réelle

Impact CPU – complexité de la requête

- Une requête simple sera rapide à traiter
- Une requête complexe sera plus longue à traiter
 - Le CPU va exécuter d'autres requêtes en même temps
 - Toutes ces requêtes vont donc subir une latence due à ce multiplexage

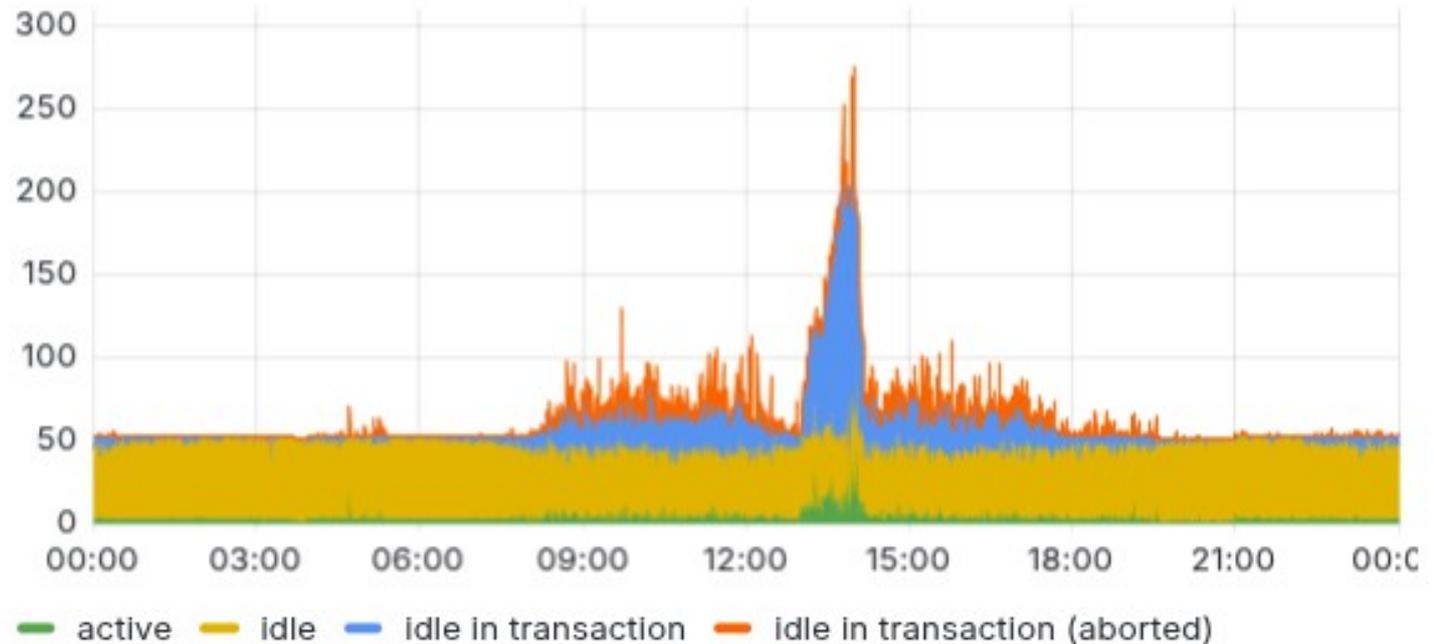
Toutes les connexions

Nombre de connexions (All)



Connexions par état

Connexions par état (All) - stacked



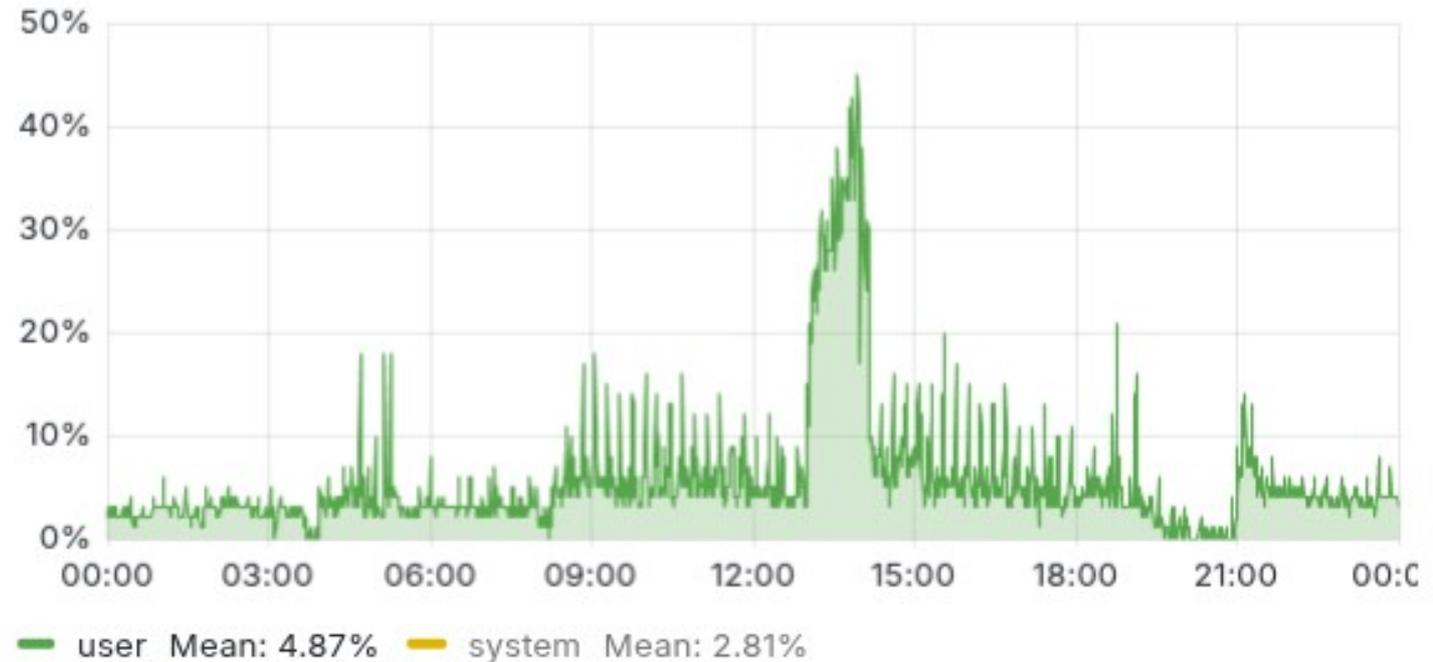
Toutes les connexions **actives**

Connexions actives (All)



Utilisation CPU

Utilisation CPU (user+sys)



Changer max_connections

- Nécessite un redémarrage
- Pourquoi ?
 - impacte la volumétrie des structures en mémoire partagée
 - mémoire partagée statique, allouée uniquement au démarrage de PostgreSQL
 - donc déconnecter tout le monde, donc diminuer sa disponibilité
 - à ce sujet, un *patch* circule pour modifier le `shared_buffers` à chaud, ce qui pourrait avoir un impact là dessus aussi
- De ce fait, la bonne configuration de ce paramètre est essentiel

Valeur basse

- Permet de consommer moins de mémoire partagée
- Permet de consacrer plus de mémoire privée par session
 - `work_mem`
 - `maintenance_work_mem`
- Problématique si pas assez haut

Deux types d'erreur

- FATAL: remaining connection slots are reserved for roles with the SUPERUSER attribute
- FATAL: sorry, too many clients already

Deux paramètres associés

- `superuser_reserved_connections`
 - Pour les utilisateurs ayant l'attribut SUPERUSER
- `reserved_connections`
 - À partir de la 16
 - Pour les utilisateurs membres du rôle `pg_use_reserved_connections`

Valeur haute

- Permet de répondre à plus de clients
 - À condition d'avoir un nombre de CPU en adéquation
- Un CPU peut exécuter plusieurs requêtes en quasi simultané
 - Si la requête reste simple
 - Plus la requête est complexe, moins le côté « simultané » sera ressenti

Surcharge d'activité ?

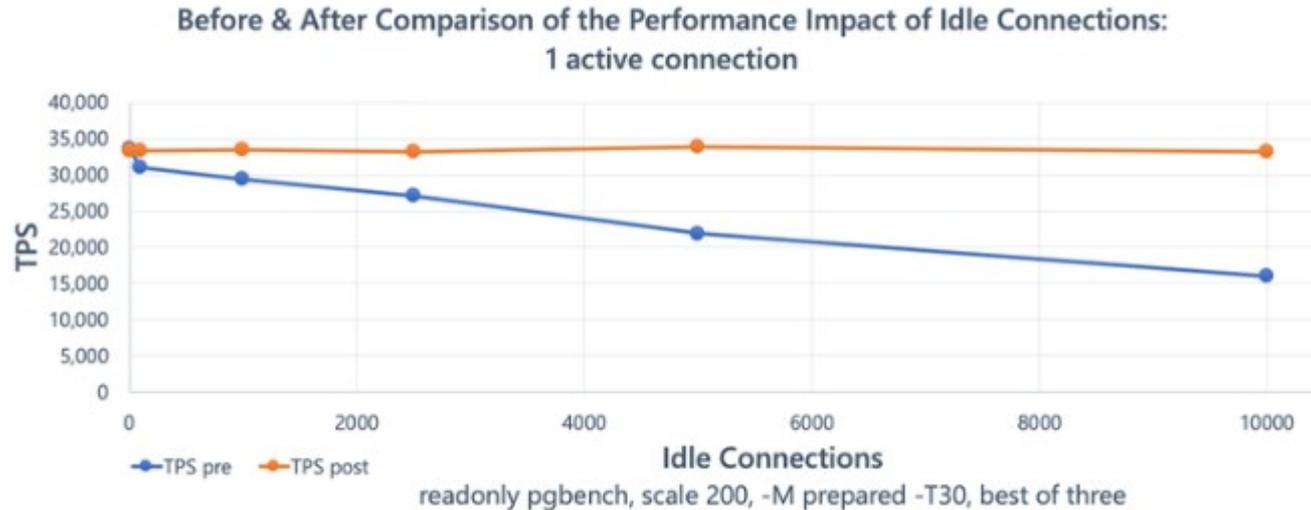
Connexions par état (All) - stacked



Petite complexité

- Une session qui n'exécute pas de requête
 - ne consomme pas de CPU
 - mais sa présence fait consommer plus de CPU aux sessions actives
- Une session active récupère des informations sur les sessions en cours
 - plus il y a de sessions, plus cette récupération d'informations prend du temps
 - moins problématique depuis la version 14

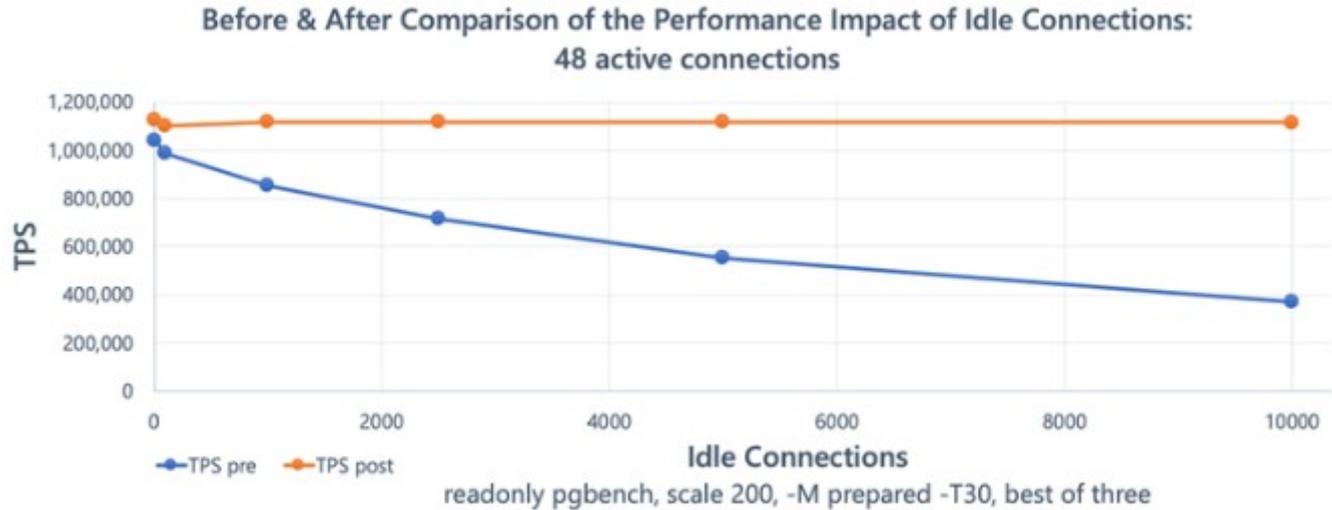
Graphe TPS vs nb connexions



Graphe provenant d'un article d'Andres Freund

Tout savoir sur max_connections

Grappe TPS vs nb connexions



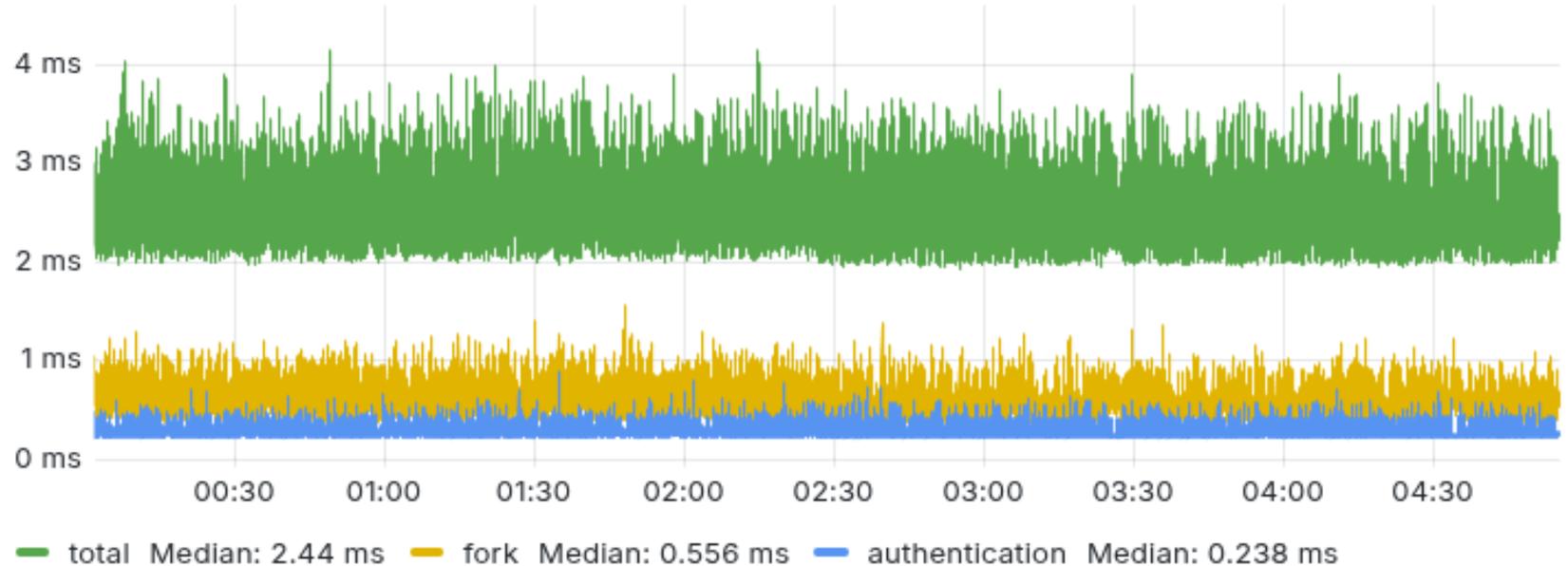
Grappe provenant d'un article d'Andres Freund

Qu'en conclure ?

- Augmenter `max_connections` n'est pas forcément une bonne idée
- Sessions actives ou inactives
 - Toutes consomment ou font consommer du CPU

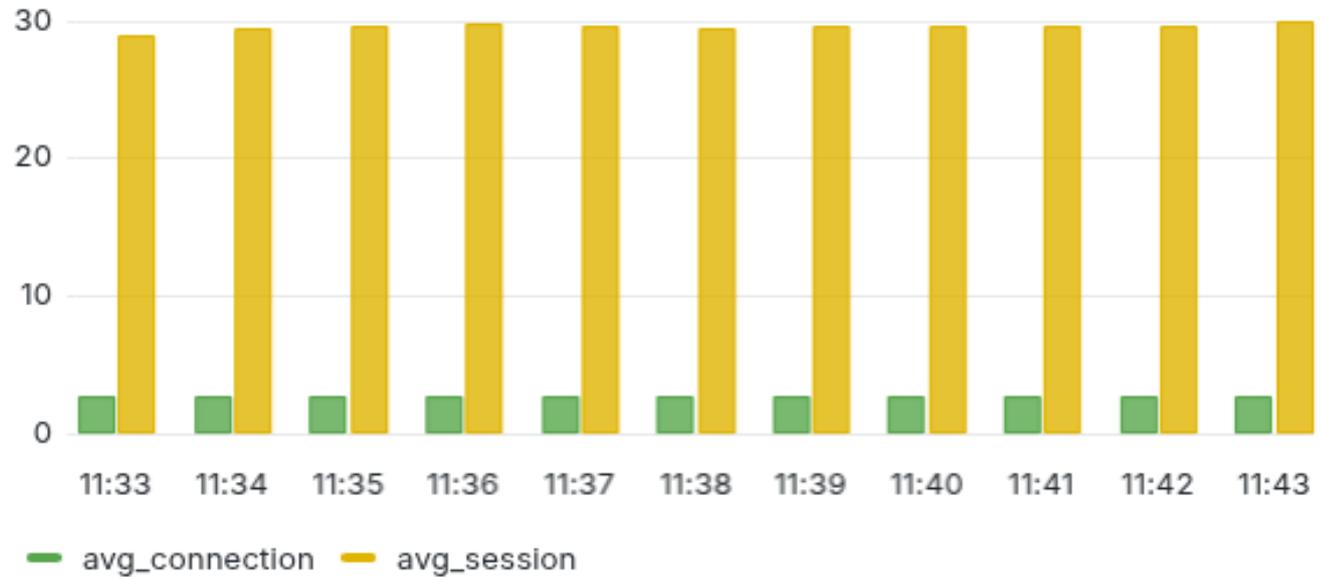
V18 beta 1 - log_connections=all

Durée des étapes d'une connexion



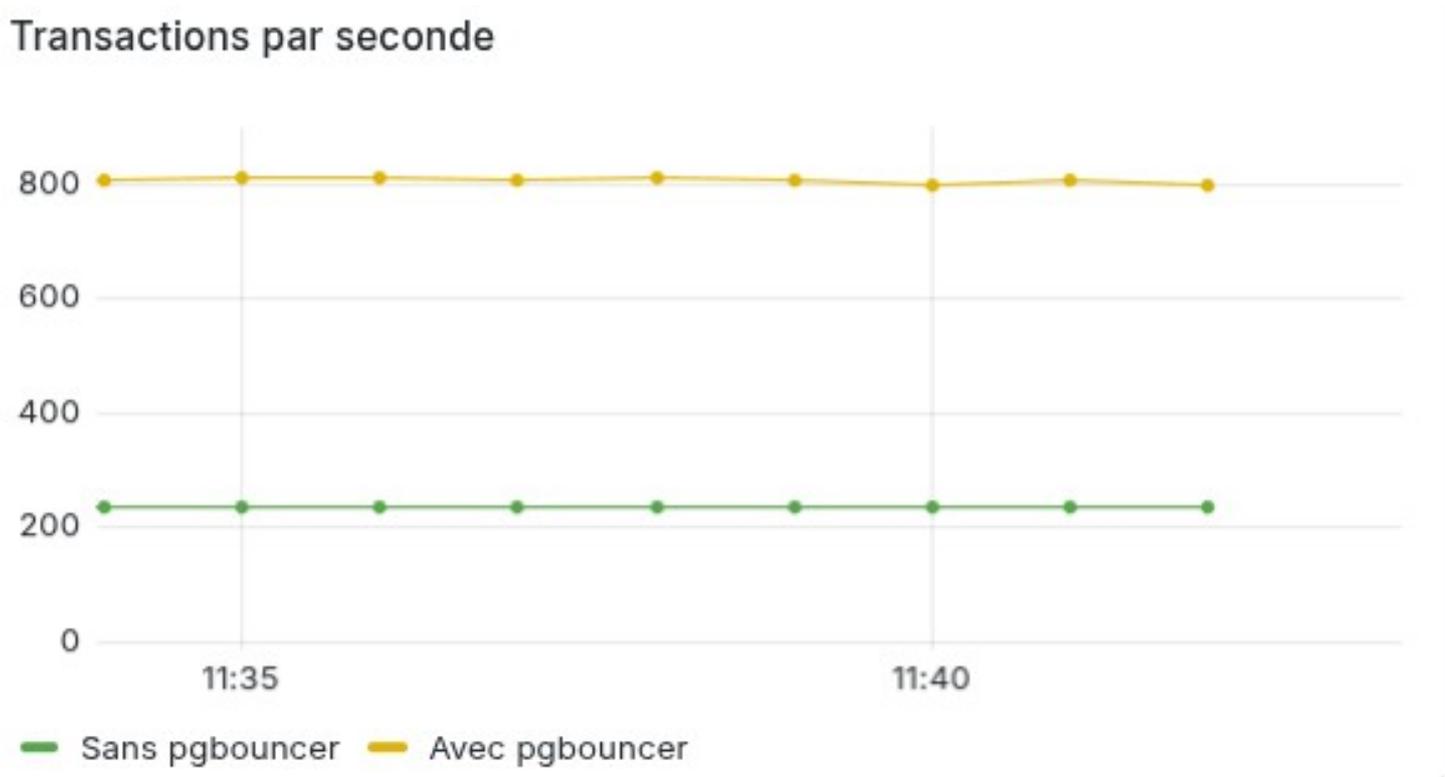
v18 beta 1 - pgbench -c 10 -C -T 600

Durée connexion/session



TPS sans et avec un pooler

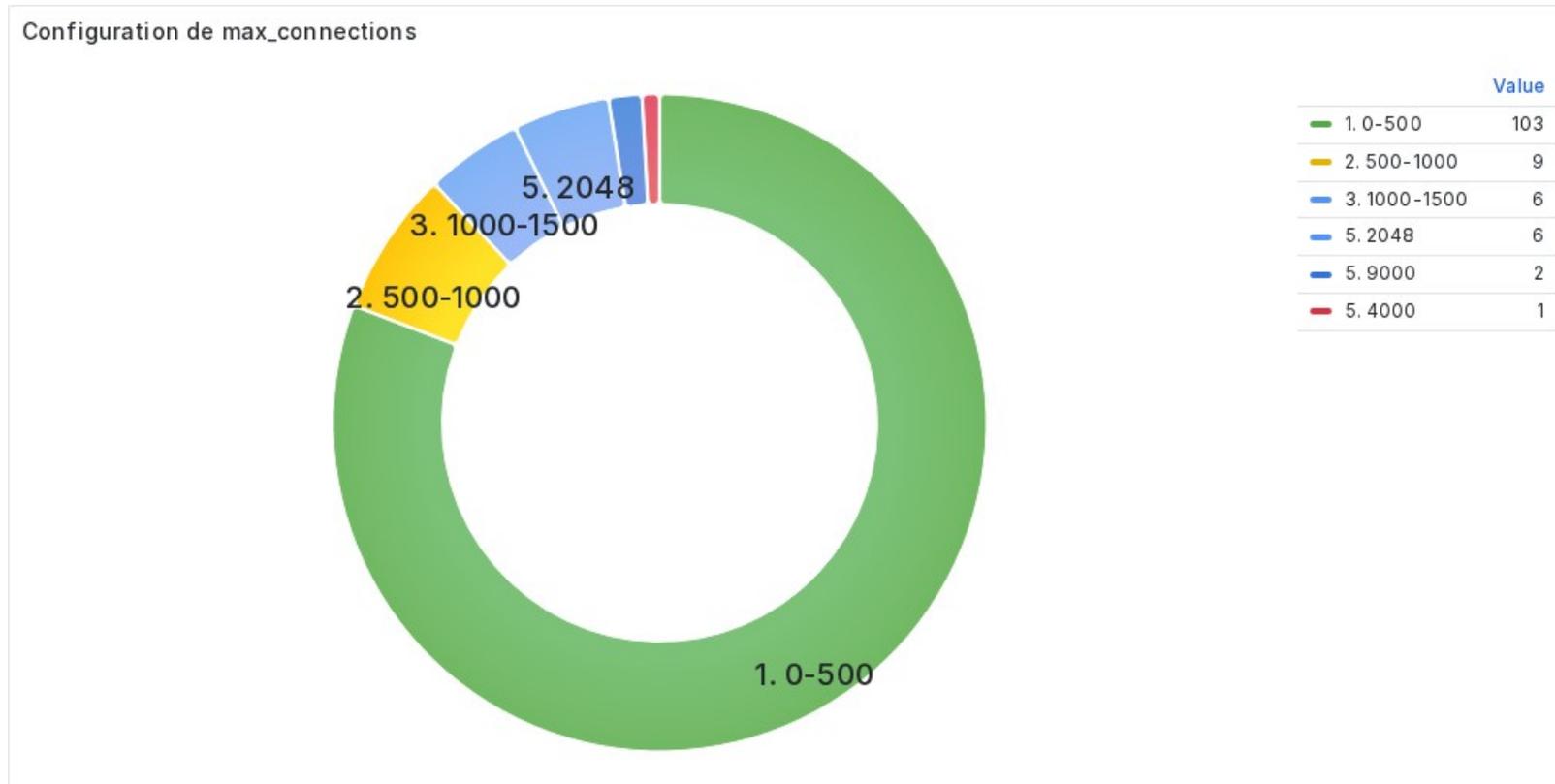
Transactions par seconde



Pour finir

- Deux petites infos supplémentaires...
- Statistiques peu représentatives sur la configuration de `max_connections`
- Différence primaire/secondaire

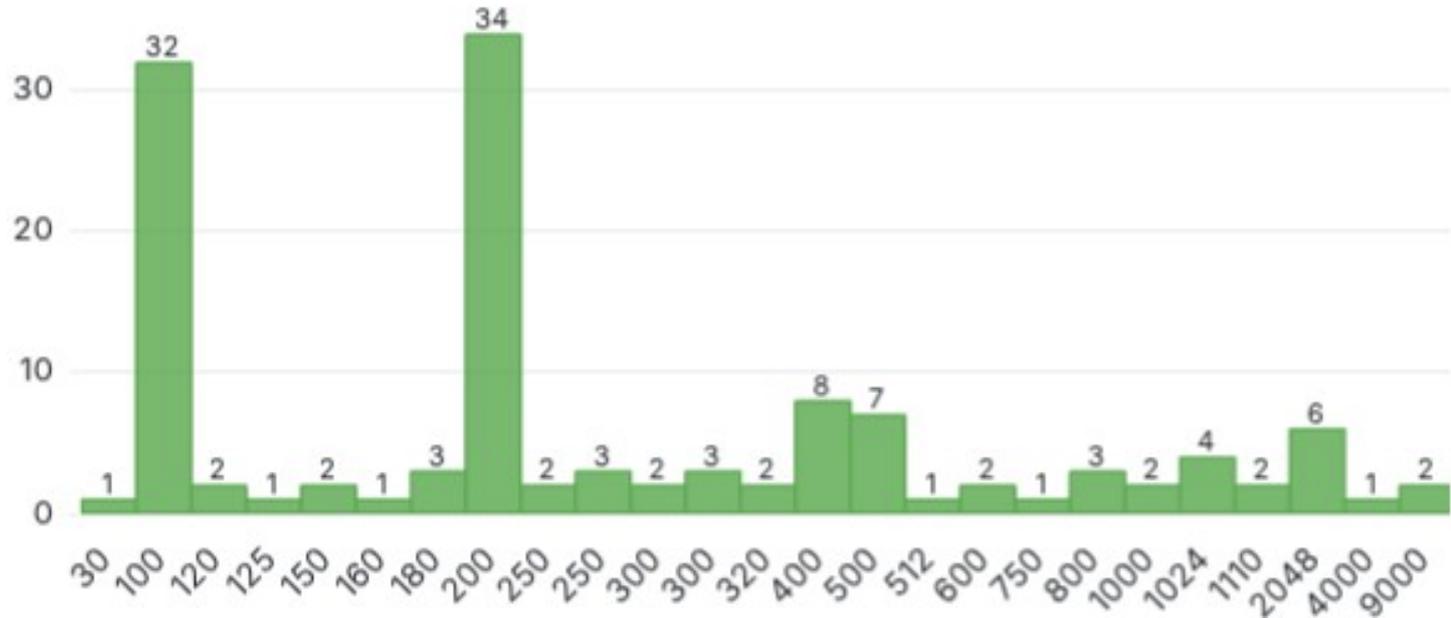
Configuration de max_connexions chez certains clients



Tout savoir sur max_connexions

En plus détaillé...

Configuration de max_connections



Différence primaire/secondaire

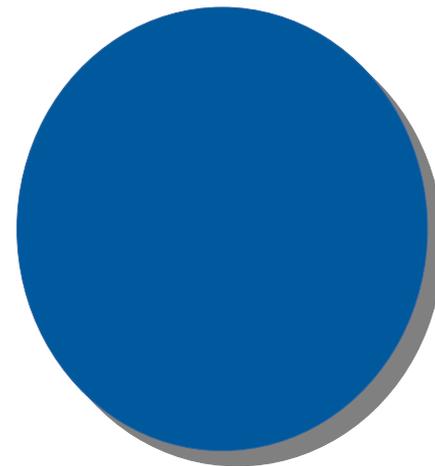
- Un primaire et un secondaire ont généralement la même configuration
- Il peut arriver que celle-ci diffère
- Attention, valeur sur le primaire \leq valeur sur le secondaire

Conclusion

- Paramètre apparemment simple
 - Mais complexe à configurer intelligemment
- N'oubliez pas l'adéquation nécessaire entre nombre de sessions et nombre de CPU
- L'utilisation d'un pooler peut aider à mieux utiliser les CPU présents
- Préférer un pooler de connexions à un `max_connections` énorme

Guillaume Lelarge

guillaume.lelarge@dalibo.com



À bientôt !