

Le partitionnement de tables démystifié

Flavio Henrique A. Gurgel

Proposition

- Penser au partitionnement
- Dans une même instance PostgreSQL
- Les mythes - vrai ou faux ?
- La décision

Votre participation est bienvenue !

Flavio Gurgel

- Passionné des données depuis 1995
- PostgreSQL depuis 1999
- Oracle, MySQL, Microsoft et DB2 font partie de mon expérience
- Brésilien en France depuis 2014
- Actif dans les communautés
- DBA et AD chez Mirakl - www.mirakl.com

Je pense à partitionner lorsque...

- j'ai des requêtes lentes
- j'ai de grosses tables
- les coeurs sont trop utilisés
- le disque est utilisé à 100%
- les temps des *I/O* sont hauts
- un développeur a eu l'idée de le faire
- le CTO a eu l'idée ou a demandé de le faire
- maman au secours, etc.

Une table d'un million de lignes

- Dois-je partitionner ?

Une table d'un million de lignes

- Dois-je partitionner ?

Faux, la taille de la table n'est pas un critère.

Une requête avec un *seqscan*

- Dois-je partitionner ?

Une requête avec un seqscan

- Dois-je partitionner ?

Faux si la requête parcourt toute la table.

Vrai si le *seqscan* parcourt une ou deux partitions, et si elle est la principale requête sur la table partitionnée.

Il s'agit d'un système OLAP

- **Dois-je partitionner ?**

Il s'agit d'un système OLAP

- Dois-je partitionner ?

Faux, parfois un système OLTP peut profiter du partitionnement aussi. Voir aussi la réponse sur des requêtes avec *seqscan*.

Les écritures sont trop lentes

- Dois-je partitionner ?

Les écritures sont trop lentes

- Dois-je partitionner ?

Faux, pour les INSERT qui seront toujours plus lents.

Vrai, pour les UPDATE et DELETE qui tombent dans le même cas des *seqscans* déjà vus.

Vrai, si on peut supprimer une partition entière parfois avec un DROP TABLE.

La table est trop gonflée (a.k.a. fragmentée, *bloated*)

- Dois-je partitionner ?

La table est trop gonflée (a.k.a. fragmentée, *bloated*)

- Dois-je partitionner ?

Vrai, quand une table est assez grosse de façon que l'autovacuum n'arrive pas à récupérer son espace.

Faux, si c'est possible de faire la maintenance d'une autre façon.

Et alors, *go* ou *no go* ?

- En 15 minutes c'est difficile de décider :)
- Il y a d'autres critères pratiques
- Analyser toujours tous les critères
- Rappel - il y a plusieurs types de requêtes
- Le but - les bases de données ne sont pas simples et **le partitionnement ne résout pas tous les problèmes de performance.**

Questions ?

MERCI !!!

fhagur@gmail.com

@fhagur