



PostgreSQL : DBaaS

Comparatif des offres Cloud

Sommaire

- Présentation
- Définition
- Les éditeurs
- Comparatif des fonctionnalités
- Comparatif des prix
- Petit guide pour choisir une offre
- Démo (en images)
- Bench
- Remerciements
- Questions

Présentation

- Thomas Poindessous
 - Directeur des opérations chez SKALE-5

- SKALE-5 : “Nous, c’est le RUN” (dans le Cloud)

- Une dizaine d’années d’expérience de prod avec des bases de données PostgreSQL.

PostgreSQL DBaaS : petite définition

- DBaaS : DataBase As A Service, base de données managée
- C'est à dire ?
 - Plus besoin d'installer le système d'exploitation
 - Plus besoin d'installer PostgreSQL
 - Plus besoin de configurer la haute disponibilité
 - Plus besoin de configurer le logiciel de sauvegarde
 - Plus besoin de configurer le monitoring
 - Plus besoin de configurer les serveurs dédiés à la lecture seule (Read Replica)
 - Plus besoin des OPS, le DBA devient autonome ! (Voire, le Dev aussi !)

Un large choix d'éditeurs de "PostgreSQL DBaaS"

- 11 éditeurs recensés !
- Bien sûr, plusieurs éditeurs majeurs de Cloud Computing : **Amazon Web Services** (AWS), **Google Cloud Computing** (GCP), **Microsoft Azure**, **Heroku** et **Alibaba Cloud**
- D'autres éditeurs + spécialisés qui proposent des offres basées sur des VM allouées chez un éditeur de Cloud Computing : **EnterpriseDB** (AWS, GCP), **Aiven** (AWS, GCP, Azure, DigitalOcean, UpCloud), **DatabaseLabs** (AWS, GCP, DigitalOcean), **Compose** (AWS, GCP, IBM Cloud), **ElephantSQL** (AWS, GCP, Azure, SoftLayer)
- Un frenchie : **OVH** mais offre réduite :(

Comparatif des fonctionnalités, 1/4

- Versions supportées
 - Au moins la 9.6, souvent aussi la 9.4 et la 9.5
 - Version 10 déjà supportée chez OVH, Aiven et ElephantSQL
- Backups et PITR
 - PITR de 7 jours jusqu'à 35 jours (AWS, Azure), pas encore de PITR chez Google
 - Backups automatiques : explicitement `pg_dump` avec stockage objet (S3) externalisé ou snapshots du serveur (et donc restauration via re-création d'un nouveau serveur)
- Monitoring
 - Monitoring système du serveur
 - Monitoring PostgreSQL : réplication, nb de connexions, hit ratio, ...
 - Monitoring des requêtes lentes (Slow Queries), des erreurs SQL

Comparatif des fonctionnalités, 2/4

- Haute Disponibilité (HA)
 - Deuxième instance en Standby (pas forcément en HotStandby, donc pas de lecture)
 - Attention à la méthode de bascule
 - Bascule DNS : impact potentiel sur les applications (AWS, ElephantSQL, Aiven ?, ...)
 - Migration IP : validée chez Google
- Performances
 - Choix des instances : de 1 CPU/1 Go RAM à
 - 40 CPU/160 Go RAM (AWS), 32 CPU/244 Go RAM (AWS)
 - 64 CPU/416 Go RAM (GCP)
 - Attention au choix du disque de stockage (SSD, IOPS) et des perfs réseau de l'instance
 - Read Replicas

Comparatif des fonctionnalités, 3/4

- Sécurité
 - SSL
 - Encryption at Rest (stockage)
- Fork : duplication simplifiée d'une base de données
 - pour la preprod, pour tester un bug de prod, pour valider une prochaine mise à jour
 - attention à la RGPD (Règlement Général pour la Protection des Données) / GDPR (General Data Protection Regulation)
- Taille des bases supportées
 - 6 To chez AWS, 10 To chez GCP

Comparatif des fonctionnalités, 4/4

- Extensions :
 - jusqu'à 50 extensions (AWS)
 - tout le monde propose PostGIS
 - certains laissent une liberté totale (EnterpriseDB, DatabaseLabs, ...)
- Accès root via la console (ssh) : EnterpriseDB, DatabaseLabs
 - on s'éloigne du Service Managé
- HA + Read Replica MultiCloud et migration entre Clouds : Aiven, ElephantSQL
- Outils ligne de commande (CLI) ou API Rest => Automatisation, intégration dans une chaîne d'intégration/déploiement (CI/CD)

Comparatif rapide des prix

- Prix : c'est la jungle, très compliqué de comparer
 - Le prix dépend du type d'instance, de la taille et performance du stockage, du nombre de données échangées en sortie du Cloud Provider.
 - Certains proposent des ristournes en échange d'une réservation longue durée (de 1 à 3 ans).
- Comparons un exemple : 8 CPU / 32 Go RAM / 100 Go de stockage

Offre	AWS	GCP (Beta)	Azure (Preview)	EnterpriseDB	Aiven	ElephantSQL	Heroku	DatabaseLabs	Compose	Alibaba
Prix en \$ par mois	602	421	640 (?)	2 000	600 (GCP) 800 (AWS)	2 000 (pour 2 To)	750 (pour 15 Go RAM)	1 700	1 200 (pour 10 Go RAM)	880 (500 en Chine)

Petit guide pour choisir une offre

- Où est mon application ?
 - Il faut essayer de rapprocher (physiquement) l'application de la base de données
 - Performances réseau donc performances applicatives
 - Coûts liés à la consommation des données échangées via le réseau
- Choix des caractéristiques de l'instance
 - Plutôt orientée CPU ou mémoire ?
 - Performances du stockage des données
 - Possibilité d'utiliser des Read Replica dans mon application ?
- Quelle durée d'indisponibilité je peux supporter ?
 - Activation ou non de la Haute Disponibilité (HA)
- Et tous les autres critères :)

Demo Time !

(en images ...)

Création d'une instance CloudSQL PG 9.6 sur GCP

- On utilise l'outil CLI appelé gcloud => principe d'automatisation (vous pouvez utiliser l'API Rest ou utiliser l'outil Terraform)
- Création d'une instance, puis modification du mot de passe de l'utilisateur postgres

```
thomas@sk5-bastion-thomas:~$ gcloud sql instances create test-pgsession --database-version=POSTGRES_9_6 --c
pu=1 --memory=4096MiB --region=europe-west1
Creating Cloud SQL instance...done.
```

```
Created [https://www.googleapis.com/sql/v1beta4/projects/sk5-formations-thomas/instances/test-pgsession].
```

NAME	DATABASE_VERSION	REGION	TIER	ADDRESS	STATUS
test-pgsession	POSTGRES_9_6	europe-west1	db-custom-1-4096	35.195.156.176	RUNNABLE

```
thomas@sk5-bastion-thomas:~$ gcloud sql users set-password postgres no-host --instance=test-pgsession --pro
mpt-for-password
Instance Password:
Updating Cloud SQL user...done.
```

```
thomas@sk5-bastion-thomas:~$
```

Création d'une instance, suite

- GCP crée une base avec une IP publique, donc on doit lister les IP autorisées à se connecter :
 - `gcloud sql instances patch test-pgsession --authorized-networks=$IP_THOMAS`
- On peut maintenant utiliser les outils habituels. Exemple avec `psql` :

```
thomas@sk5-bastion-thomas:~$ psql -h 35.195.156.176 -U postgres
Password for user postgres:
psql (9.6.5, server 9.6.1)
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES128-GCM-SHA256, bits: 128, compression: off)
Type "help" for help.

postgres=> create database thomas;
CREATE DATABASE
postgres=> \q
thomas@sk5-bastion-thomas:~$
```

C'est tout ?

- Maintenant, on active la HA :

```
thomas@sk5-bastion-thomas:~$ gcloud beta sql instances patch test-pgsession-2 --availability-type REGIONAL
The following message will be used for the patch API method.
{"project": "sk5-formations-thomas", "name": "test-pgsession-2", "settings": {"availabilityType": "REGIONAL"}}
Patching Cloud SQL instance...done.
Updated [https://www.googleapis.com/sql/v1beta4/projects/sk5-formations-thomas/instances/test-pgsession-2].
thomas@sk5-bastion-thomas:~$
```

- Et puis, on va dupliquer cette instance (en gardant la HA sur l'instance dupliquée :

```
thomas@sk5-bastion-thomas:~$ gcloud sql instances clone test-pgsession-2 test-pgsession-clone
Cloning Cloud SQL instance...done.
Created [https://www.googleapis.com/sql/v1beta4/projects/sk5-formations-thomas/instances/test-pgsession-clone].
```

NAME	DATABASE_VERSION	REGION	TIER	ADDRESS	STATUS
test-pgsession-clone	POSTGRES_9_6	europe-west1	db-custom-1-4096	35.205.45.94	RUNNABLE

```
thomas@sk5-bastion-thomas:~$
```

Le Bench !

Éléments du bench

- GCP vs AWS
- Instance : 8 CPU / 32 Go RAM / 100 Go stockage
- `pgbench -i -s 640 (10 Go) (en 5 minutes)`
- Plusieurs lancements depuis une instance GCP ou AWS à 4 CPU :
 - `pgbench -c 10 -j 2 -t 10000`
 - `tps = 1787.870969` contre `tps = 1740.672974` (6 ms de latence)
 - `pgbench -c 20 -j 4 -t 10000`
 - `tps = 3214.458687` contre `tps = 2462.975566` (6 ms de latence contre 8 ms)
 - `pgbench -c 40 -j 4 -t 10000`
 - `tps = 3077.839444` contre `tps = 3110.634773` (12 ms de latence)
 - `pgbench -c 90 -j 4 -t 10000`
 - `tps = 2457.264196` contre `tps = 2858.612946` (30 ms de latence)

Pour tester: les offres gratuites !

- ElephantSQL : 20 Mo de stockage
- Heroku : 10 000 lignes
- AWS : 1 petite instance avec 20 Go de stockage pendant 1 an
- GCP : 300\$ de crédits, valable 12 mois
- Azure : 200\$ de crédits, valable 1 mois

Conseils de vieux sage

- Passez dans le Cloud, en utilisant des services managés
- Automatisez et testez tout ce que vous pouvez
- Embauchez un DBA ou au moins, commandez des audits (code, perf) régulièrement
- Former vos développeurs au SQL et à PostgreSQL
- Offrez le livre de Dimitri Fontaine “Mastering PostgreSQL in Application Development”
 - <https://masteringpostgresql.com/>

Remerciements

- Dalibo
- SKALE-5
- La communauté PostgreSQL et les entreprises associées
- François G., mon DBA PostgreSQL préféré

Questions ?