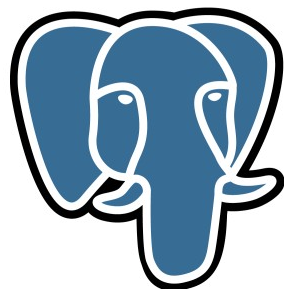


PostgreSQL, PostGIS

Les usages pour la sécurité intérieure

PostgreSQL



PostGIS



Présentations

Eric Pommereau

- Ingénieur des SIC, ancien policier (PJ Paris)
- Chef de section cartographie au ST(SI)²
- Animateur de la CIMI (Communauté Info...)

Vincent Laborie

- Ingénieur des SIC
- Veille SGBD
- Bidouilleur postgresql

Partie I, PostGIS dans la sécurité intérieure :

- Le ST(SI)²
- Le SIG de la sécurité intérieure
- Les usages de PostGIS
- Refonte IDIC-SI : Mysql > PostgreSQL

Partie II, pourquoi un proxy-cache OSM sous postgresql ?

- Houston on a un problème...
- Solution(s) ?
- Le présent...
- Le futur...

Contexte : le ST(SI)²

*Le Service des Technologies et des Systèmes
d'Information de la Sécurité Intérieure*



La sécurité intérieure :

- Police nationale, effectifs : 145 197
- Gendarmerie nationale, effectifs : 98 155
- Répartition sur (environ) 5000 sites

Au Ministère de l'Intérieur :

- Plusieurs DSI (DSIC, PP, ST(SI)²)
- Réseau privé
- Cadre de cohérence technique

Missions du ST(SI)² :

- Définition de la stratégie pour la sécurité intérieure
 - Pour les SIC
 - Pour la SSI
- Coordonner les services SIC de proximité
- Animer la politique d'innovation technologique
- Piloter et animer le Service de Développement Logiciel et le CNAU (support)
- Concevoir et conduire des projets pour la Police et la Gendarmerie.

Le SIG de la Sécurité Intérieure

Le socle SIG, une plate-forme cartographique



Le SIG :

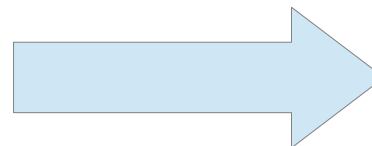
- Une plateforme, le socle SIG
 - Marché public : CapGemini / ESRI.
 - Hébergement à Rosny-sous-Bois (CTGN)
- Fonctionnalités :
 - Diffusion de données : vecteurs et rasters (OGC)
 - Services de localisation (LBS)
 - Principe d'accès par API (indépendance backend)
- Rôle : « servir » les projets de la sécurité intérieure
- Impératifs : disponibilité, performance
- Stockage : File GDB, ShapeFile et **PostGIS**

PostGIS : les usages

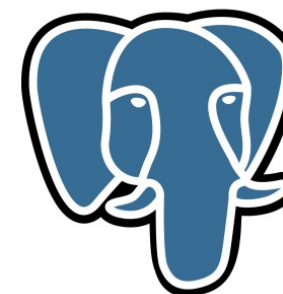
- **Services du socle SIG**
 - Gestion AVL (géolocalisation des véhicules)
 - Services rendus par l'API (géo-référencement)
- **Activité de la section cartographie**
 - Gestion du référentiel cartographique
 - Enrichissement statistique
 - Analyse des AVL
- **Applications**
 - QGIS (intégration pour le déploiement)
 - Application de sectorisation infra communale
 - IDIC-SI

Migration MySQL > PostgreSQL

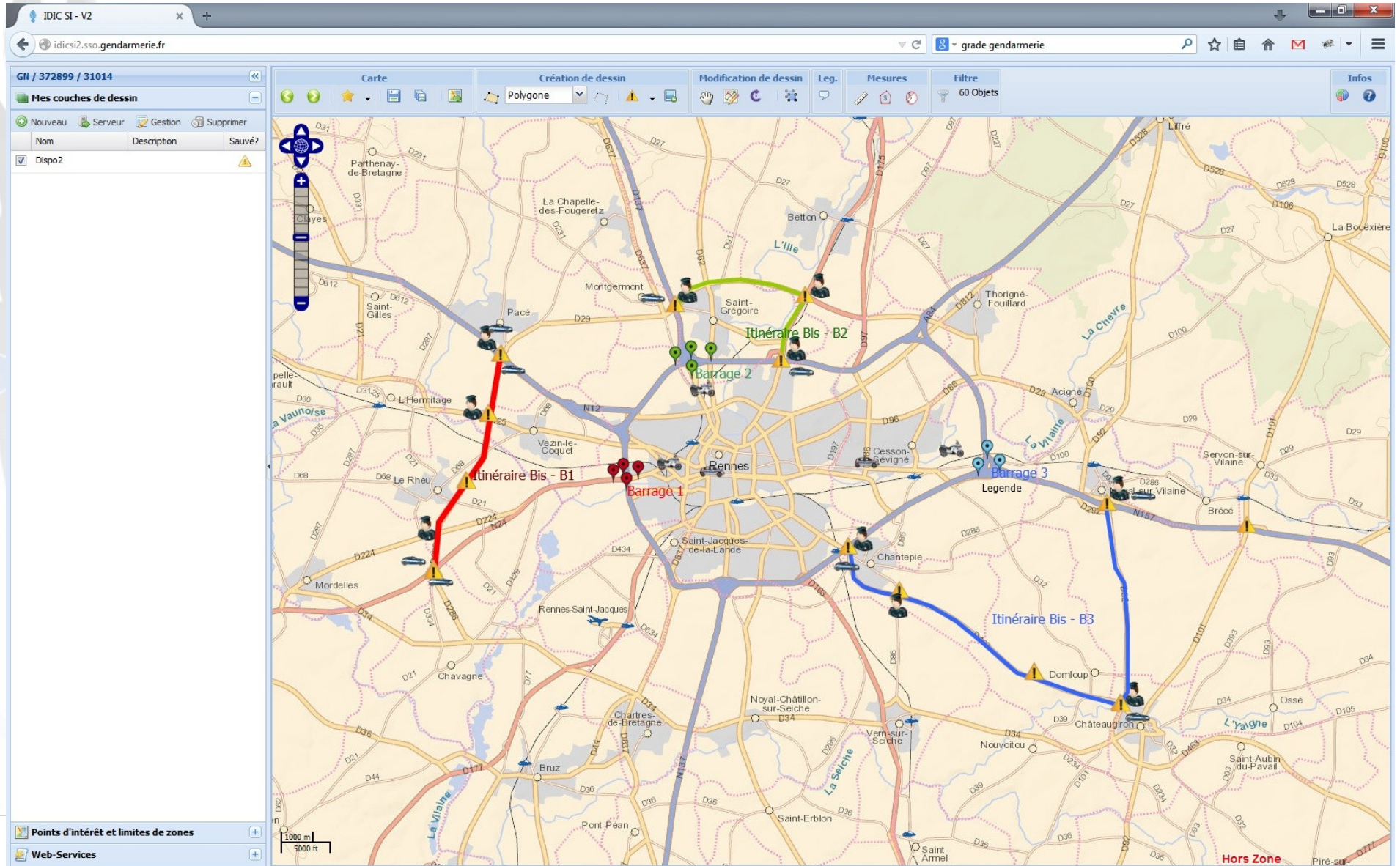
IDIC-SI, retour d'expérience sur la refonte d'une application web de dessin cartographique



PostgreSQL



IDIC-SI – Retour d'expérience



IDIC-SI – Retour d'expérience

Caractéristique : ouverte à tout policier/gendarme.

Principales fonctions :

- Afficher des fonds de carte (OSM, Ortho, « maison »)
- Afficher des données de référence
- Dessiner (polygones, lignes, ponctuels)
- Enregistrer, partager les couches de données
- Utiliser des services du socle SIG (LBS)
- Importer et exporter : CSV, KML

IDIC-SI – Retour d'expérience

Exemples d'utilisations :

- **Planification et conduite d'événements**
 - 70ème anniversaire du débarquement
 - Tour de France
 - Suivi de visites officielles
- **Préparation d'interventions (PJ, sécurité publique...)**
- **Aide à l'enquête**
 - Illustration de scène crime
 - Disparitions
- **Projection statistique** (visualisation)

IDIC-SI – Retour d'expérience

→ Un saut vers l'inconnu

- L'équipe :
 - Un chef de projet
 - Un développeur
 - Un stagiaire
- Pas (ou peu) de connaissances
 - En postgresSQL
 - En postGIS

→ **De très fortes attentes...**

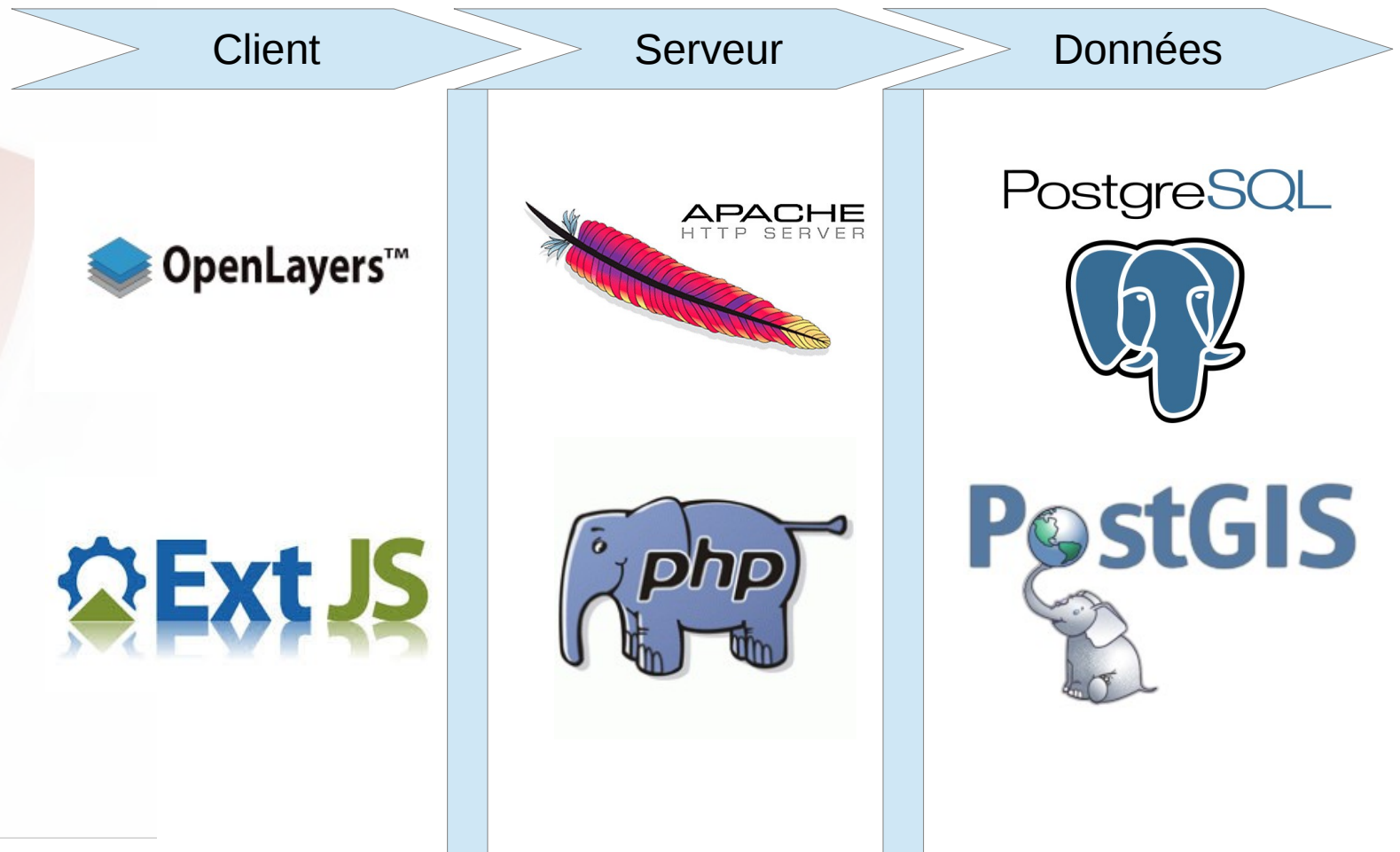
IDIC-SI – Retour d'expérience

Objectifs de la refonte :

- Garder la maîtrise : développement interne
- Unifier de deux instances Police / Gendarmerie
- Absorber la charge potentielle (Police + Gendarmerie)
- Mettre à jour les composants logiciels
- Faciliter le développement de nouvelles fonctionnalités (conception à revoir)

IDIC-SI – Retour d'expérience

Choix de la pile technique :

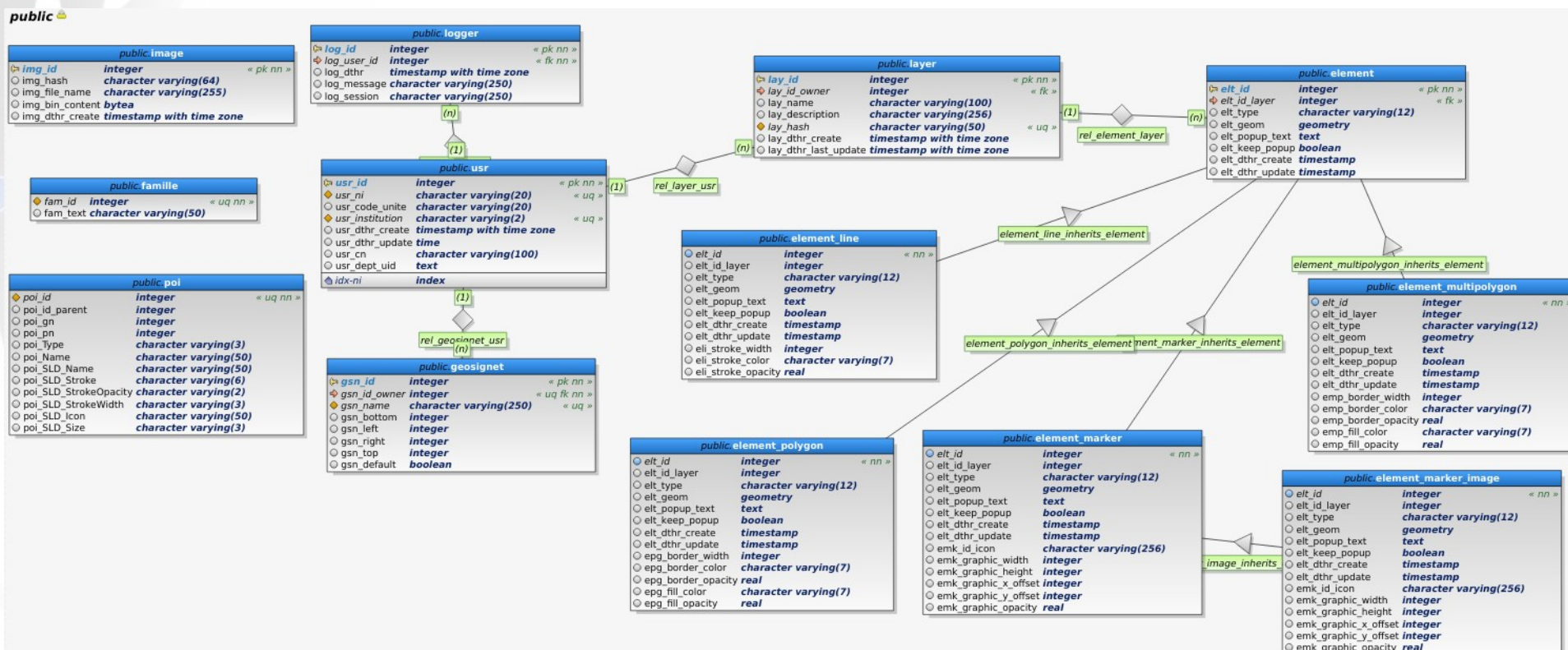


IDIC-SI – Retour d'expérience

- Initialement (Mysql)
 - Pas de stockage « SIG »
 - Stockage au format KML (statique)
 - Peu de contraintes BDD, pas de transaction
- PostgreSQL
 - Mise en oeuvre :
 - Contraintes (PK, check, unique, FK)
 - Héritage de tables
 - Support des transactions (PHP/PDO)
 - PostGIS
 - Reprojection (différents basemaps)
 - Vérification des géométries (ST_IsValid)

IDIC-SI – le choix de PG/postGIS

Modélisation (travail commun avec Vincent)



IDIC-SI – le choix de PG/postGIS

- **Statistiques**
 - **Applicatives**
 - **Utilisateurs : 44 000**
 - **Couches : 27 000**
 - **Stockage SGBD : 900 Mo**
- **L'avenir :**
 - **Poursuivre les itérations courtes (MEP tous les 2 mois), retour très positif du terrain.**
 - **Suivre les évolutions du Socle SIG**
 - **Tirer davantage partie de PostGIS**

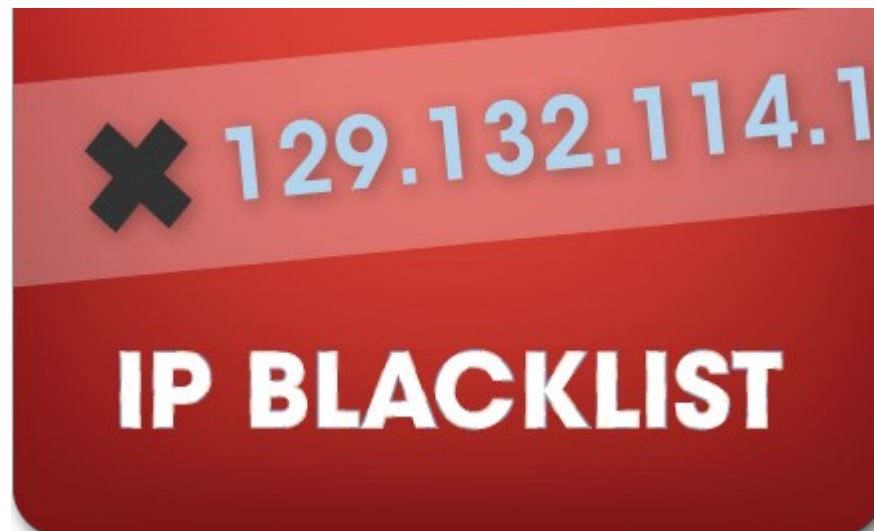
Le bilan

- **Gros challenge mais aucun regret**
 - Objectif atteint : stabilité et tenue en charge
 - Produit réalisé en interne et maîtrisé
- **Nécessité d'accompagner la montée en compétence des développeurs peu formés à PG (formations, ateliers...)**
- **Expertise interne très appréciable (merci Vincent)**
- **Coordination nécessaire avec les exploitants (compétences, possibilités techniques...)**

Pourquoi un proxy-cache des tuiles osm ?

Problème → idée(s) → solution

- Et oui, pourquoi ?



Bulk downloading is strongly discouraged. Do not download tiles unnecessarily.

http://wiki.openstreetmap.org/wiki/Tile_usage_policy



Mais pourquoi pas une usine OSM ?

→ du temps + des ressources

Et si on faisait un cache des tuiles en interne ?

- Machine : VM Linux 4 coeurs – 4 Go RAM initialement (16 Go maintenant)
- Langage : Go (<http://golang.org/>) (226 lignes)
- POC : (<https://github.com/communaute-cimi/rosm>)
- Postgresql 9.4.4 + extensions :
 - pg_buffercache
 - pg_prewarm
 - pg_stat_statements
 - pgstattuple
- Pgcluu (<http://pgcluu.darold.net/>)
- Pgbadger (<http://dalibo.github.io/pgbadger/>)
- Pg_activity (https://github.com/julmon/pg_activity/)
- Pg_view (https://github.com/zalando/pg_view)

- **Un usage conséquent (moyenne sur 90 jours) :**

Lecture : 14445 tuiles / heure

Mise à jour : 623 tuiles / heure

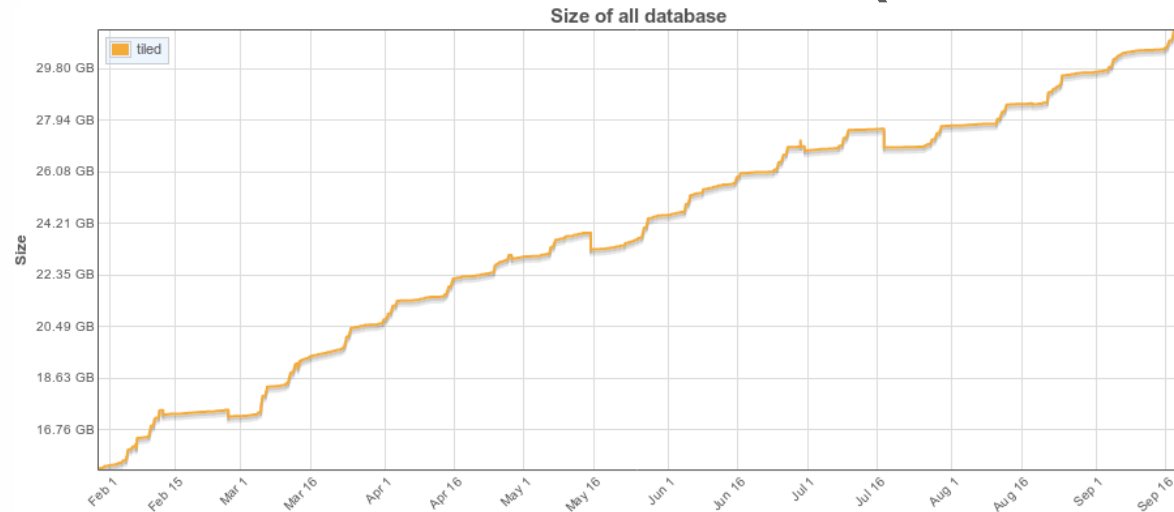
Insertion : 334 tuiles / heure

Temps de restitution d'une tuile : 1,5 ms

Temps d'écriture d'une tuile : 12 ms

Temps moyen d'une mise à jour : 17 ms

3,6M de tuiles en base – 31 Go (+ 2 Go/mois)



Le futur ?

- Passage en usine OSM
- Passage en machine physique (IO...)
- Ajouter un système token

Pour nous suivre sur Twitter : [@communaute_cimi](https://twitter.com/communaute_cimi)





Merci !!

Des questions ?