

CatalogueR les données d'un serveur PostgreSQL/Postgis sans (trop d') effort

Lisa SMAH* Edouard MORIN† Juliette ENGELAERE-LEFEBVRE‡

Résumé (171 mots)

La Direction régionale de l'environnement de l'aménagement et du logement des Pays de la Loire dispose d'un important patrimoine de données, enrichi en continu, utilisé par une cinquantaine d'agents pour produire des connaissances territoriales dans le domaine de l'environnement, des risques, du logement et des transports. Pour en faciliter l'accès et en améliorer leur usage, un catalogue de données a été développé avec RShiny et Golem. Le concept : aucunes fiches de métadonnées à rédiger, juste des commentaires de table et de champs directement saisis en base, lors de l'enregistrement d'une table.

Pour la tenir à jour, un traitement est exécuté quotidiennement par notre runner gitlab. Il découvre les tables en base, prépare les données à visualiser dans le catalogue et les déploie dans l'application. {CatalogueR} est le package Shiny/Golem développé pour répondre à ce besoin. L'adhérence à notre environnement est très faible (quelques lignes de paramétrage) et le code source est public.

Cette présentation décrit le fonctionnement de l'application, sa recette de fabrication, les conditions de réussite et les perspectives.

Mots-clefs (3 à 5) : Package - DataOps - Reproductibilité - Shiny - Institution publique

Développement

La Direction régionale de l'environnement de l'aménagement et du logement des Pays de la Loire est un service déconcentré du ministère de la transition écologique. Il dispose d'un important patrimoine de données, enrichi en continu, utilisé par une cinquantaine d'agents pour produire des connaissances territoriales dans le domaine de l'environnement, des risques, du logement et des transports. Pour en faciliter l'accès et en améliorer leur usage, plusieurs solutions commerciales ou open source dédiées au catalogue de données ont été envisagées.

Jugées trop coûteuses en temps à passer et/ou financièrement, un catalogue de données {catalogueR} a été développé en interne avec RShiny et Golem, le temps de l'apprentissage de Lisa Smah.

Le concept : aucunes fiches de métadonnées à rédiger ! Elles ne seraient jamais toutes à jour. Juste des commentaires de table et de champs directement saisis en base, lors de l'enregistrement ou de la modification d'une table, pour n'avoir qu'une source de vérité : le serveur de base de données.

Grâce aux fonctionnalités CI/CD de gitlab, un traitement de mise à jour / déploiement est exécuté quotidiennement par notre runner gitlab. Il découvre les tables en base grâce à {datalibaba}, notre package maison de liaison entre un serveur de bases de données PostgreSQL et R. Il prépare ensuite les données à visualiser dans le catalogue et les déploie dans l'application.

{CatalogueR} est le package Shiny/Golem développé pour répondre à ce besoin et est l'objet de cette intervention.

L'adhérence de ce package à notre environnement est très faible (quelques lignes de paramétrage); la charte graphique propre aux services de l'état peut facilement être enlevée et le code source est public : <https://gitlab-forge.din.developpement-durable.gouv.fr/dreal-pdl/csd/catalogueR>. Aussi, l'idée de l'intervention

*DREAL Pays de la Loire, apprentie

†DREAL Pays de la Loire, edouard.morin@developpement-durable.gouv.fr

‡DREAL Pays de la Loire, juliette.engelaere@developpement-durable.gouv.fr

est de montrer le résultat, c'est-à-dire de décrire le fonctionnement de l'application du point de vue d'un utilisateur, puis de montrer comment se l'approprier en tant que développeur R.

Cette présentation fera un détour sur le package `{datalibaba}`, pierre angulaire de la préparation des données. `{datalibaba}` a été conçu pour faciliter les interactions entre R et un serveur PostgreSQL. Les identifiants de connexion sont masqués dans le `.Renvi` pour pouvoir déposer ses scripts sur une forge git sans exposer ses secrets. La syntaxe de lecture est unifiée pour l'utilisateur Riste qui ne se préoccupe plus de savoir quelle fonction utiliser si les données ont une composante spatiale afin de retrouver les propriétés géographiques du dataset lu, ni de comment déclarer correctement le pointeur "schéma" "table" pour éviter que la table ne finisse dans le schema public. . .

Bref, toute une série de petites choses qui facilite l'administration de données pour un Riste dont le SQL n'est pas la spécialité.

En guise de conclusion, la présentation abordera les conditions de réussite d'un tel projet et les perspectives d'évolution du produit.

Références

- <https://gitlab-forge.din.developpement-durable.gouv.fr/dreal-pdl/csd/catalogueR>
- <https://gitlab-forge.din.developpement-durable.gouv.fr/dreal-pdl/csd/datalibaba>