

GeoNature et Docker

Présentation des nouveautés et de perspectives



Sommaire

1. Qu'est-ce que Docker ?
2. Mise en place de GeoNature avec Docker : l'existant
3. Développement de GeoNature avec Docker
4. Flexibilisation de l'instanciation de GeoNature avec Docker
5. Tour de Table

Bonus. Installation d'un module externe sur une instance GN dockerisé

Qu'est-ce que Docker ?



- **Docker** est un système de conteneurisation d'application
 - Pratique pour gérer plusieurs applications avec des dépendances différentes
 - Un “gestionnaire de processus” != “Machine virtuelle”
 - Une installation d'applications (la plupart du temps) simplifiées !
 - Grâce au Dockerfile -> partition de l'installation de l'application
 - Et plein d'autres !

Pourquoi utiliser Docker pour installer GeoNature ?

Administration

- Une mise à jour simplifiée (2 commandes

```
docker compose pull  
docker compose up -d
```

- Déploiement de plusieurs instances facilité

Développement

- Travailler dans un environnement très proche d'un environnement de prod
- Aucun effet "ça marche sur ma machine"

Installation de GeoNature avec Docker

Utilisation de GeoNature Docker Services

GeoNature Docker Services

- Docker-compose permet d'orchestrer différents conteneurs (BDD, backend, frontend, etc..) qui permettent le fonctionnement d'une application
- GDS propose un `docker-compose` pour installer : *GeoNature*, *UsersHub*
- S'appuie sur traefik

```
● ● ●
# Récupération des fichiers nécessaires
git clone https://github.com/PnX-SI/GeoNature-Docker-services
cd GeoNature-Docker-services

# Génération de la configuration
cp .env.sample .env
./init-config.sh

# Démarrage de l'application de GeoNature
docker compose up -d

# GeoNature est accessible sur https://localhost/geonature
# UsersHub est accessible sur https://localhost/usershub
```

Développement de GeoNature avec Docker

Utilisation du docker compose de dev

Deux constats

1. Coût d'entrée dans GeoNature important

- plusieurs paquet et installation pré-requis ;
- pas de guide unique et clair ;
- plusieurs instances cohabitent difficilement.

2. Difficile de gérer les BDD en développement :

- Une longue installation ;
- Quid du développement sur plusieurs versions de GN à la fois ?

Deux solutions (qui se combinent très bien !)

- La BDD pré-peuplée ;
- Le docker compose de dev ;

GeoNature-db : Une base de données GeoNature pré-peuplée

- Build à chaque nouvelle release de GN (depuis la 2.15.2) ;
- Contient les mêmes données qu'une BDD GN post-init ;
- Quasiment instantanée ;
- Plusieurs images peuvent facilement cohabiter

```
docker run -d \  
  --rm -p 5432:5432 \  
  --name geonature-db \  
  ghcr.io/pnx-si/geonature-db-extra:latest
```

Un environnement prêt en quelques minutes

- Clone du projet ;
- Décommenter les lignes dans le fichier `.env` ;
- Le premier lancement peut prendre une dizaine de minutes ;
- Instance disponible sur <https://localhost/geonature/#/>.



```
git clone https://github.com/PnX-SI/GeoNature-Docker-services.git
cd GeoNature-Docker-services
make submodule_init
make dev
```

- Passage en mode DEV dans le fichier .env

```
COMPOSE_FILE=docker-compose.yml:docker-compose-dev.yml
```

- Les images sont builds au démarrage du docker-compose ;
- Les Dockerfile ou leurs targets sont spécifiques au dev :
 - Backend : installation de GN et module en mode *pip -e* ;
 - Frontend: installation dynamique des modules au démarrage, lancement via *npm run start*.
- Exposition de la BDD et de l'interface de Traefik ;
- Montage des sources en volume.

Flexibilisation de l'instanciation de GeoNature avec Docker

Évolution de GeoNature Docker Services

Faciliter le passage entre une installation classique avec un déploiement docker

- Possibilité de rester sur sa base de données ;

Outil de *reverse-proxy*

- Avoir la possibilité de choisir une autre solution que traefik ;
- Utiliser la solution existante dans son infrastructure.

Modularité

- Rendre facile l'installation d'un autre service de la suite GN comme Geonature-Atlas.

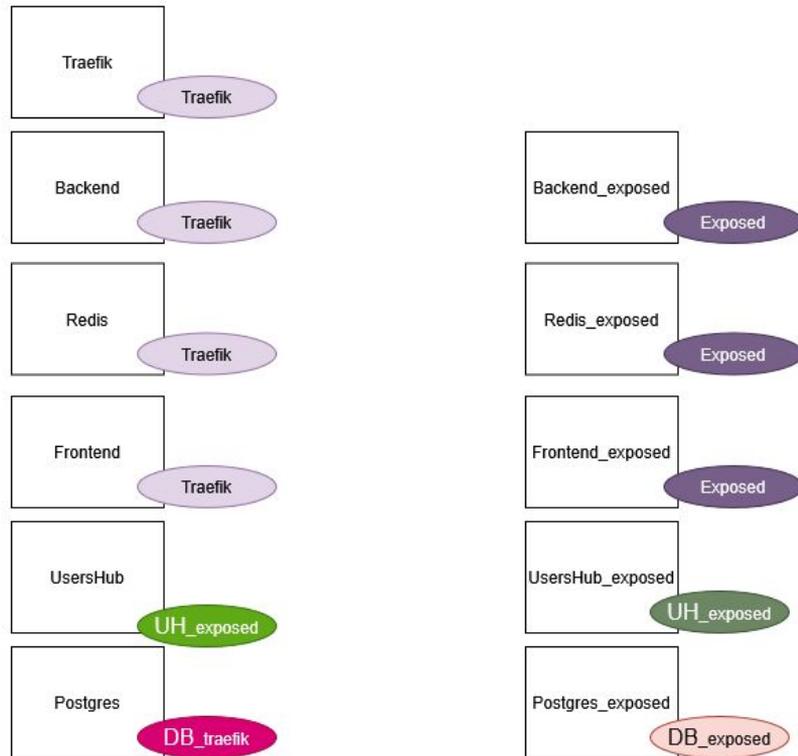
Retour sur Docker Compose

```
services:
  service-1:
    image: postgres:latest
    ports:
      - "5432:5432"
    volume:
      postgres_data:
    ...
  service-2:
    image: geonature-backend:latest
    ports:
      - "8000:8000"
    ...
networks:
  gn:
volumes:
  postgres_data:
```

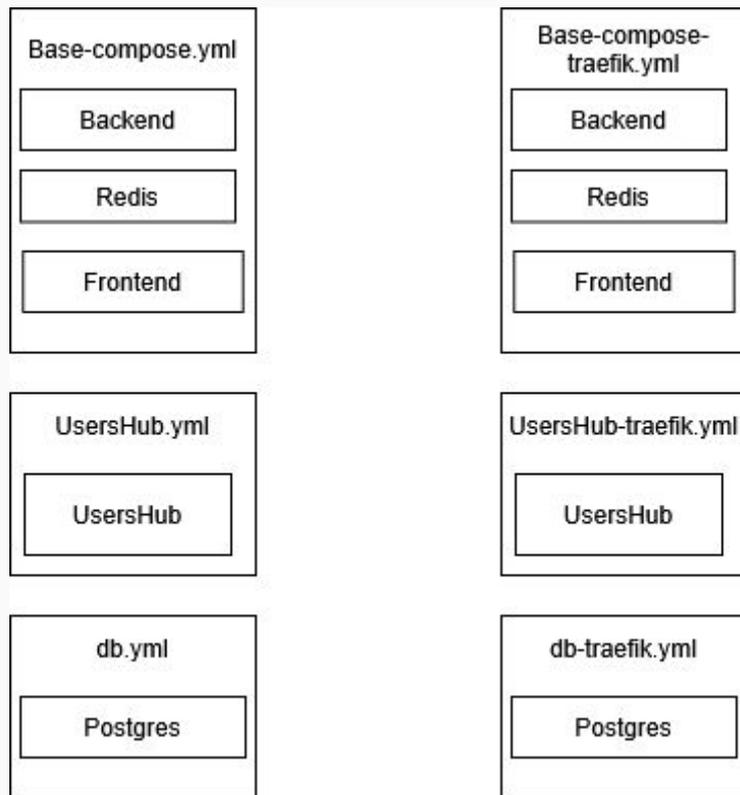
Utilisation des **profiles** de DockerCompose

```
services:
  db:
    image: postgres:latest
    ports:
      - "5432:5432"
    volume:
      postgres_data:
    ...
  api:
    image: geonature-backend:latest
    ports:
      - "8000:8000"
  usershub:
    image: usershub:latest
    ports:
      - "5001:5001"
    profiles: [with_usershub]
    ...
networks:
  gn:
volumes:
  postgres_data:
```

Compose



Surcharge de DockerCompose



- Même mécanisme qu'en dev ;
- Permet de modifier la définition d'un service en plus d'en ajouter un ;
- Problème !
 - Multiplication des composes.

Proposition d'une solution mixte

Compose-base.yml

Backend

Redis

Frontend

UsersHub 

Postgres 

Compose-traefik.yml

Traefik

Backend

Frontend

UsersHub 

Utilisation avec le fichier .env:

```
COMPOSE_FILE=compose-base.yml:compose-  
traefik.yml  
COMPOSE_PROFILES=db,usershub
```

OU

Utilisation avec le client Docker

```
docker compose up -f compose-base.yml  
-f compose-traefik.yml --profile db  
--profile usershub
```

Démo

Travail prévu

- Terminer le POC du nouveau GeoNature Docker Services;
- Ajouter de GeoNature-Atlas ;
- Éprouver le docker de dev et ajout de fonctionnalités (BDD pré-peuplées, cypress, ...) ;
- Ajouter dockerize/wait-for-it dans les images dockers GN ;
- Mettre à jour la documentation de développement.

Perspectives

- Ajouter des versions des bdd pré-peuplées avec plus de données ;
- Faciliter l'installation de modules externes ?
- Améliorer la gestion des variables d'environnements ?

Tour de table

Quelles sont vos besoins ?
Quelles évolutions ?

Merci pour votre attention !

BONUS. Installation d'un module externe

Actuellement en prod:

- Il faut rebuild l'image de frontend ET de back avec le nouveau module.

Actuellement en dev:

- Il faut relancer le docker front ;
- Il faut rebuild le backend avec modification du dockerfile.

Faut-il rendre ça plus simple ? Quels modules embarqués dans le docker compose fourni ?

- POC
<https://github.com/PnX-SI/GeoNature-Docker-services/pull/60>
- Ajout de GeoNature-Atlas
<https://github.com/PnX-SI/GeoNature-Docker-services/pull/51>
- Ajout de module externe dans une instance dockerisée
<https://github.com/PnX-SI/GeoNature-Docker-services/pull/51>