
Georiviere

Release 0.9.5

Makina Corpus

Nov 08, 2021

INSTALLATION CONFIGURATION

1	Install	1
1.1	Requirements	1
1.2	Install	1
1.3	Update	2
2	Configuration	3
3	Local Environment	5
4	Documentation	7
5	Publication	9
5.1	CI	9
5.2	Release	9
5.3	Docker image	9
6	CHANGELOG	11
6.1	0.9.5 (2021-11-08)	11
6.2	0.9.4 (2021-11-05)	11
7	Authors	13
8	Indices and tables	15

INSTALL

1.1 Requirements

- You need docker installed. Docker-compose is recommended in the configuration below.
- **Optional** : if you want to use external database, prepare a postgresql 10+ postgis2.5+ database with postgis and postgis_raster enabled, and a dedicated user.

You can use external database by commenting postgres container and volume references in docker-compose.yml, and set variables

- POSTGRES_HOST
- POSTGRES_PORT
- POSTGRES_USER
- POSTGRES_PASSWORD
- POSTGRES_DB
- You can use external nginx proxy. Edit provided nginx conf file and comment nginx references in docker-compose.yml. Fix web:8000 to 127.0.0.1:8000 in nginx.conf.

1.2 Install

- Download [zip package](#)
- Unzip it where you want

```
unzip install.zip  
cd georiviere
```

- Prepare environment variables

```
mv .env.dist .env
```

-> Set all required values

- Init default var folder

```
docker-compose run --rm web bash -c "exit"
```

- Set at least variables *SRID*, *DEFAULT_STRUCTURE_NAME*, and *SPATIAL_EXTENT* in var/conf/custom.py (as geotrek overlay, these settings should be set BEFORE database initialization)

- Init database and project config

```
docker-compose run --rm web update.sh
```

- Create your super user

```
docker-compose run --rm web ./manage.py createsuperuser
```

- Launch stack

```
docker-compose up
```

1.3 Update

- Read [release notes](#) about bugfix, news and breaking changes.
- Backup your data (database and var folder)
- Pull latest image

```
docker-compose pull
```

- Run post update script

```
docker-compose run --rm web update.sh
```

- Relaunch you docker-compose stack

```
docker-compose down  
docker-compose up
```

CONFIGURATION

LOCAL ENVIRONMENT

- Configuration

To get local environment working, we recommend to use a custom domain, as 'georiviere.local'. Define it in your /etc/hosts.

Copy the env dist file

```
cp .env.dist .env
```

Set required values, for postgres database access

- Init database:

```
docker-compose run --rm web ./manage.py migrate
```

- Create user:

```
docker-compose run --rm web ./manage.py createsuperuser
```

- Run:

```
docker-compose up
```

- Launch tests :

```
docker-compose run --rm web ./manage.py test
```

- With coverage :

```
docker-compose run --rm web coverage run ./manage.py test
docker-compose run --rm web coverage report -m
```

- Dependencies :

- Manage all project dependencies with pip-tools
- Use included pip-tools to generate requirements (python version should match georiviere version)

- Global dependencies :

- set global dependency in requirements.in

```
docker-compose run --rm web pip-compile
```

pip-tools does not upgrade any package by default. Package is upgrade only if new dependency require another version that already fixed in requirements.txt file.

To upgrade a package, run :

```
docker-compose run --rm web pip-compile --upgrade-package django==3.1.*
```

Development packages are separated in dev-requirements.in. dev-requirements.txt depends on requirements.txt. Each time you run pip-compile, dev-requirements.txt is updated. Each time you run pip-compile for requirements.txt, you should run pip-compile for dev-requirements.txt.

```
docker-compose run --rm web pip-compile dev-requirements.in
```

Warning: Geotrek is used as main library of this project. Sub-dependencies are not yet managed in geotrek setup.py. When you update geotrek, you should update requirements according geotrek dependencies versions.

DOCUMENTATION

We use sphinx doc and sphinx-rtd-theme.

Requirements are included.

To compile and test documentation on local environment, run :

```
docker-compose run --workdir /opt/georiviere-admin/docs --rm web make html
```


PUBLICATION

5.1 CI

- Each edition runs a CI build.
- All Georiviere-admin maintainers can review or merge Pull Requests.
- First time contributor not in maintainer team can request to be added. Maintainers can accept its first pull request to allow CI build.

5.2 Release

To release a new version :

- set version to georiviere/VERSION file.
- set changelog infos in docs/changelog.rst
- push or merge to master
- Go to <https://github.com/Georiviere/Georiviere-admin/releases>.
- Click on “Draft a new release”
- set new tag according older ones.
- Copy / paste changelog for version in release notes.
- In the end, CI publish publish new docker image to github packages.

5.3 Docker image

- Docker image is published after each release in Georiviere github repository: `ghcr.io/georiviere/georiviere-admin:latest`

CHANGELOG

6.1 0.9.5 (2021-11-08)

- Improve documentation
- Improve README, maintainers and brand mark policy

6.2 0.9.4 (2021-11-05)

- First code publication

CHAPTER
SEVEN

AUTHORS

- Emmanuelle Helly
- Timothée de Montety
- Jean-Etienne Castagnede

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`