Mise à jour de GeoNature

Comme indiquée dans la documentation de GeoNature : `La mise à jour de GeoNature consiste à télécharger sa nouvelle version dans un nouveau répertoire, récupérer les fichiers de configuration et de surcouche depuis la version actuelle et de relancer l'installation dans le répertoire de la nouvelle version.`

Ressources

- Doc mise à jour de GeoNature

Procédure

- Commencer par réaliser des sauvegardes de l'instance "*web-srv*" et vérifier le bon fonctionnement des sauvegardes de la base de données Postgresql.
- Idéalement, récupérer la dernière archive sauvegardée de la base de données en local et charger la dans Postgresql pour vérifier le bon fonctionnement de l'archive.

Créer un snapshot du volume "<sinp>-geonature-postgresql"

Cette procédure est désormais inutile car la base de données est hébergée sur le disque principal du serveur ; les volumes n'étant pas assez rapide !

- Sur l'instance "*db-srv*" stoper Postgresql : `sudo systemctl stop postgresql`
- Via l'interface d'OVH :
  - détacher le volume "<sinp>-geonature-postgresql"
  - créer le snapshot. **ATTENTION** : l'interface d'OVH semble afficher en permanence "PENDING", il faut forcer le renouvellement de la page pour voir le changement d'etat.
  - attacher à nouveau le volume à l'instance "*db-srv*"
- Sur l'instance "*db-srv*" relancer Postgresql : `sudo systemctl start postgresql`

Désactivation du service GeoNature

- Sur l'instance "*web-srv*" :
  - Pour les versions supérieur à GeoNature 2.4.2 et le fonctionnement à partir de 2022 :
    - Passer le site en maintenance longue avec la commande :

```
mv
~/www/maintenance/geonature/maintenance.disable
~/www/maintenance/geonature/maintenance.enable
```

    - Si Systemd est utilisé pour GeoNature l'Atlas, désectiver les services ainsi :

```
systemctl stop geonature taxhub usershub atlas
```

        - Si Supervisor est toujours utilisé, le désactivé comme indiqué ci-dessous.

- Pour les versions antérieurs à GeoNature 2.4.2 et le fonctionnement antérieur à 2022 :
    - Basculer GeoNature et l'Atlas en mode maintenance :
        - `nginx_dissite geonature.conf ; nginx_ensite geonature_maintenance.conf ; service nginx reload`
        - `nginx_dissite atlas.conf ; nginx_ensite atlas_maintenance.conf ; service nginx reload`
    - Arrêter les services de GeoNature de *Supervisor* avant de faire les mises à jour : `supervisorctl stop geonature2 atlas taxhub usershub2`

Mise à jour côté "web-srv" - GeoNature

- Exporter la dernière version de GeoNature dans une variable d'env locale à la session : `export GNV=$(curl -s https://api.github.com/repos/PnX-SI/GeoNature/releases/latest | grep tag_name | cut -d\" -f4)`
    - Vérifier la version en question : `echo "${GNV}"` *
- Se placer dans le dossier *dwl* de l'utilisateur *geonat* : `cd ~/dwl/`
- Télécharger l'archive : `wget https://github.com/PnX-SI/GeoNature/archive/${GNV}.zip -O geonature_v${GNV}.zip`
- Décompresser l'archive dans le dossier *www* de l'utilisateur *geonat* : `unzip geonature_v${GNV}.zip -d ./`
- Renommer le dossier au format GeoNature (⇒ uniformité): `mv GeoNature-${GNV} ~/www/geonature_v${GNV}`
- Se rendre dans le dossier *www* : `cd ~/www/`
- Pour faciliter la migration et la compatibilité avec les scripts GeoNature créer les liens symbolique suivant :
    - `cd ~/www; rm -f geonature_old; ln -s geonature_v<ancienne-version> geonature_old`
    - Seulement nécessaire la première fois :
        - `cd ~/; rm -f geonature_old; ln -s ~/www/geonature_old geonature_old`
        - `cd ~/; rm -f geonature; ln -s ~/www/geonature geonature`
- Copier les fichiers de config de l'ancien dossier vers le nouveau :
    - le fichier de paramètres pour les scripts d'installation et Gunicorn : `cp ~/www/geonature_old/config/settings.ini ~/www/geonature_v${GNV}/config/settings.ini`
    - le fichier de config du backend Python de GeoNature : `cp ~/www/geonature_old/config/geonature_config.toml ~/www/geonature_v${GNV}/config/geonature_config.toml`
    - Ajouter la modification décrite ci-dessous dans le fichier *gunicorn_start.sh* : `vi ~/www/geonature_v${GNV}/backend/gunicorn_start.sh`
- Vérifier l’existence de nouveaux paramètres de config :
    - Dans *settings.ini* : `diff ~/www/geonature_old/config/settings.ini.sample ~/www/geonature_v${GNV}/config/settings.ini.sample`

    - Dans *geonature_config.toml* : `diff ~/www/geonature_old/config/geonature_config.toml.sample ~/www/geonature_v${GNV}/config/geonature_config.toml.sample`
- Modifier le lien symbolique pour pointer vers la nouvelle version : `cd ~/www; rm -f geonature; ln -s geonature_v${GNV} geonature`
- Consulter les éventuelles notes de version spécifiques décrites au niveau de chaque version : https://github.com/PnX-SI/GeoNature/releases
    - **ATTENTION** : Avant de lancer un script, l'éditer et corriger les chemins vers l'installation de GeoNature
    - Sur l'instance "*web-srv*" exécuter tous les scripts en lien avec l'interface.

Mise à jour côté "web-srv" - UsersHub

- Exporter la dernière version de UsersHub dans une variable d'env locale à la session : `export UHV=$(curl -s https://api.github.com/repos/PnX-SI/UsersHub/releases/latest | grep tag_name | cut -d\" -f4)`
    - Vérifier la version en question : `echo "${UHV}"`
- Se placer dans le dossier *dwl* de l'utilisateur *geonat* : `cd ~/dwl/`
- Télécharger l'archive : `wget https://github.com/PnX-SI/UsersHub/archive/${UHV}.zip -O usershub_v${UHV}.zip`
- Décompresser l'archive dans le dossier *www* de l'utilisateur *geonat* : `unzip usershub_v${UHV}.zip -d ~/www/`
- Se rendre dans le dossier *www* : `cd ~/www/`
- Renommer le dossier au format GeoNature (⇒ uniformité): `mv UsersHub-${UHV} usershub_v${UHV}`
- Pour faciliter la migration et la compatibilité avec les scripts de migration créer les liens symbolique suivant :
    - `cd ~/www; rm -f usershub_old; ln -s usershub_v<ancienne-version> usershub_old`
    - Seulement nécessaire la première fois :
        - `cd ~/; rm -f usershub; ln -s ~/www/usershub usershub`
        - `cd ~/; rm -f usershub_old; ln -s ~/www/usershub_old usershub_old`
- Copier les fichiers de config de l'ancien dossier vers le nouveau :
    - le fichier de paramètres pour les scripts d'installation et Gunicorn : `cp ~/www/usershub_old/config/settings.ini ~/www/usershub_v${UHV}/config/settings.ini`
    - le fichier de config du backend Python : `cp ~/www/usershub_old/config/config.py ~/www/usershub_v${UHV}/config/config.py`
- Vérifier l’existence de nouveaux paramètres de config :
    - Dans *config.py* : `diff ~/www/usershub_old/config/config.py.sample ~/www/usershub_v${UHV}/config/config.py.sample`
    - Dans *settings.ini* : `diff ~/www/usershub_old/config/settings.ini.sample ~/www/usershub_v${UHV}/config/settings.ini.sample`
- Modifier le lien symbolique pour pointer vers la nouvelle version : `cd ~/www; rm -f usershub; ln -s usershub_v${UHV} usershub`
- Consulter les éventuelles notes de version spécifiques décrites au niveau de chaque version :

https://github.com/PnX-SI/UsersHub/releases
    - **ATTENTION** : Avant de lancer un script, l'éditer et corriger les chemins vers l'installation de GeoNature
    - Sur l'instance "*web-srv*" exécuter tous les scripts en lien avec l'interface.

Mise à jour côté "web-srv" - TaxHub

- Exporter la dernière version de TaxHub dans une variable d'env locale à la session : `export THV=$(curl -s https://api.github.com/repos/PnX-SI/TaxHub/releases/latest | grep tag_name | cut -d\" -f4)`
    - Vérifier la version en question : `echo "${THV}"`
- Se placer dans le dossier *dwl* de l'utilisateur *geonat* : `cd ~/dwl/`
- Télécharger l'archive : `wget https://github.com/PnX-SI/TaxHub/archive/${THV}.zip -O taxhub_v${THV}.zip`
- Décompresser l'archive dans le dossier *www* de l'utilisateur *geonat* : `unzip taxhub_v${THV}.zip -d ~/www/`
- Se rendre dans le dossier *www* : `cd ~/www/`
- Renommer le dossier au format GeoNature (⇒ uniformité): `mv TaxHub-${THV} taxhub_v${THV}`
- Pour faciliter la migration et la compatibilité avec les scripts de migration créer les liens symbolique suivant :
    - `cd ~/www; rm -f taxhub_old; ln -s taxhub_v<ancienne-version> taxhub_old`
    - Seulement nécessaire la première fois :
        - `cd ~/; rm -f taxhub; ln -s ~/www/taxhub taxhub`
        - `cd ~/; rm -f taxhub_old; ln -s ~/www/taxhub_old taxhub_old`
- Copier les fichiers de config de l'ancien dossier vers le nouveau :
    - le fichier de paramètres pour les scripts d'installation et Gunicorn : `cp ~/www/taxhub_old/settings.ini ~/www/taxhub_v${THV}/settings.ini`
    - le fichier de config du backend Python : `cp ~/www/taxhub_old/config.py ~/www/taxhub_v${THV}/config.py`
    - le fichier de config du frontend Javascript : `cp ~/www/taxhub_old/static/app/constants.js ~/www/taxhub_v${THV}/static/app/constants.js`
- Transférer les médias uploadés dans l'ancienne version vers la nouvelle : `cp -aR ~/www/taxhub_old/static/medias/ ~/www/taxhub_v${THV}/static/medias/`
- Vérifier l'existence de nouveaux paramètres de config :
    - Dans *config.py* : `diff ~/www/taxhub_old/config.py.sample ~/www/taxhub_v${THV}/config.py.sample`
    - Dans *settings.ini* : `diff ~/www/taxhub_old/settings.ini.sample ~/www/taxhub_v${THV}/settings.ini.sample`
    - Dans *constants.js* : `diff ~/www/taxhub_old/static/app/constants.js.sample ~/www/taxhub_v${THV}/static/app/constants.js.sample`
- Modifier le lien symbolique pour pointer vers la nouvelle version : `cd ~/www; rm -f taxhub; ln -s taxhub_v${THV} taxhub`
- Consulter les éventuelles notes de version spécifiques décrites au niveau de chaque version : https://github.com/PnX-SI/UsersHub/releases

    - **ATTENTION** : Avant de lancer un script, l'éditer et corriger les chemins vers l'installation de GeoNature
    - Sur l'instance "*web-srv*" exécuter tous les scripts en lien avec l'interface.

Mise à jour côté "db-srv"

- Si les notes de version indiquent qu'il faut exécuter un script de migration de base de données suivre les étapes suivantes :
    - Synchroniser le dossier GeoNature de "web-srv" vers "db-srv" : `rsync -av -e "ssh -p <port-ssh-db>" /home/geonat/www/ geonat@db-<region>-sinp:/home/geonat/www/`
- Se connecter sur l'instance "*db-srv*" en tant que *geonat* : `ssh geonat@db-<region>-sinp`
    - Seulement nécessaire la première fois, créer des liens symboliques depuis la racine du home car ils sont utilisés par les scripts de migration :
        - `cd ~/; rm -f geonature_old; ln -s ~/www/geonature_old geonature_old`
        - `cd ~/; rm -f geonature; ln -s ~/www/geonature geonature`
        - `cd ~/; rm -f usershub; ln -s ~/www/usershub usershub`
        - `cd ~/; rm -f usershub_old; ln -s ~/www/usershub_old usershub_old`
        - `cd ~/; rm -f taxhub; ln -s ~/www/taxhub taxhub`
        - `cd ~/; rm -f taxhub_old; ln -s ~/www/taxhub_old taxhub_old`
    - Exporter les dernières versions de appli GeoNature dans des variables d'environnement locale à la session :
        - GeoNature : `export GNV=$(curl -s https://api.github.com/repos/PnX-SI/GeoNature/releases/latest | grep tag_name | cut -d\" -f4)`
        - UsersHub : `export UHV=$(curl -s https://api.github.com/repos/PnX-SI/UsersHub/releases/latest | grep tag_name | cut -d\" -f4)`
        - TaxHub : `export THV=$(curl -s https://api.github.com/repos/PnX-SI/TaxHub/releases/latest | grep tag_name | cut -d\" -f4)`
        - Vérifier : `echo "GNV: ${GNV} ; UHV: ${UHV} ; THV: ${THV}"`
    - Suivez les notes de version indiquant l'exécution de scripts liés à la base de données :
        - GeoNature : https://github.com/PnX-SI/GeoNature/releases
        - UsersHub : https://github.com/PnX-SI/UsersHub/releases
        - TaxHub : https://github.com/PnX-SI/TaxHub/releases

Réactivation du service GeoNature

- Sur l'instance "*web-srv*" :
    - Démarrer les services GeoNature de *Supervisor* : `supervisorctl start geonature2 atlas taxhub usershub2`
    - Désactiver le mode maintenance de GeoNature et de l'Atlas :
        - `nginx_dissite geonature_maintenance.conf ; nginx_ensite geonature.conf ; service nginx reload`
        - `nginx_dissite atlas_maintenance.conf ; nginx_ensite atlas.conf ; service nginx reload`

Installation d'une branche hors releases

GeoNature

Lors de la phase de développement de nouvelle fonctionnalités sur GeoNature, il peut être intéressant de pouvoir les tester en ligne. Pour ce faire, il est important que la branche en question se base sur la dernière version installée sur le serveur de GeoNature. Sinon, il risque d'y avoir des problème de compatibilité avec la base de données. **ATTENTION** : Si la branche nécessite une mise à jour de la base de données, bien vérifier que cette mise à jour est réversible !

Voici la procédure à suivre pour mettre en place un telle branche sur "*web-srv*" :

- Exporter le nom de la branche dans une variable d'env locale à la session. C'est le nom utilisé pour les fichiers il faut donc y remplacer les éventuels "/" par "-" : `export GNB="<branche>"`
- Indiquer la version de Geonature sur laquelle se base les développements
  - Pour voir la version, en local dans le dossier du code de GeoNature contenant les développements : `cat VERSION`
  - Récupérer aussi les 7 derniers caractères du SHA1 du dernier commit de la branche à récupérer
  - Exporter la version de GeoNature optenu avec *cat* en la concaténant avec les 7 premiers caractères du commit séparé par "-" . Ex. : `export GNV="2.10.0.dev0-5208519"`
- **ATTENTION** : si des modifications ont été faites dans des lib dont dépend GeoNature (*utils_flask_sqla*) faire les modif à la main une fois la migration effectuée dans le *venv*. Par exemple, pour *utils_flask_sqla* dans `~/www/geonature/backend/venv/lib/python3.7/site-packages/utils_flask_sqla`. Cela ne devrait plus être nécessaire avec les versions 2.8.2+ car les dépendances sont inclues via `pip install -r requirements`
- Vérifier les versions en question : `echo "Branche: ${GNB} ; GeoNature : ${GNV}"`
- Se placer dans le dossier *dwl* de l'utilisateur *geonat* : `cd ~/dwl/`
- Télécharger l'archive : `wget https://github.com/PnX-SI/GeoNature/archive/<branche>.zip -O "geonature_${GNB}_v${GNV}.zip"`
  - Il est aussi possible d'indiquer un SHA1 de commit pour l'URL : `https://github.com/PnX-SI/GeoNature/archive/<sha1>.zip`
- Décompresser l'archive dans le dossier *dwl* de l'utilisateur *geonat* : `unzip geonature_${GNB}_v${GNV}.zip -d ~/dwl/`
- Renommer le dossier au format GeoNature (⇒ uniformité), dans le nom de la branche les "/" sont remplacés par des "-" : `mv GeoNature-${GNB} geonature_${GNB}_v${GNV}`
- Déplacer le dossier au format GeoNature dans *www/* : `mv geonature_${GNB}_v${GNV} ~/www/`
- Se rendre dans le dossier *www* : `cd ~/www/`
- Copier les fichiers de config de la version de prod vers le dossier de la branche :
  - le fichier de paramètres pour les scripts d'installation et Gunicorn : `cp ~/www/geonature/config/settings.ini ~/www/geonature_${GNB}_v${GNV}/config/settings.ini`
  - le fichier de config du backend Python de GeoNature : `cp ~/www/geonature/config/geonature_config.toml ~/www/geonature_${GNB}_v${GNV}/config/geonature_config.toml`

    - Configuration de Gunicorn :
        - Pour GN > 2.8.1, voir la section "Surcouchage des fichiers Systemd de GeoNature (GeoNature > v2.8.1)" pour la configuration de Gunicorn. Gunicorn ne nécessite plus de modification spécifique des fichiers de GN, mais nous devons copier (ou le créer) le fichier `environ` de paramétrage : `cp ~/www/geonature/environ ~/www/geonature_${GNB}_v${GNV}/environ`
        - Pour GN =< 2.8.1, le fichier de démarrage de Gunicorn : `cp ~/www/geonature/backend/gunicorn_start.sh ~/www/geonature_${GNB}_v${GNV}/backend/gunicorn_start.sh`
- Vérifier l'existence de nouveaux paramètres de config :
    - Dans *settings.ini* : `diff ~/www/geonature/config/settings.ini.sample ~/www/geonature_${GNB}_v${GNV}/config/settings.ini.sample`
    - Dans *geonature_config.toml* : `diff ~/www/geonature/config/geonature_config.toml.sample ~/www/geonature_${GNB}_v${GNV}/config/geonature_config.toml.sample`
- Compléter/modifier les paramètres de config en fonction des besoins de la branches dans les fichiers :
    - *settings.ini* : `vi ~/www/geonature_${GNB}_v${GNV}/config/settings.ini`
    - *geonature_config.toml* : `vi ~/www/geonature_${GNB}_v${GNV}/config/geonature_config.toml`
- Copier les fichiers existant des composants personnalisables du frontend :
    - Sauvegarder le dossier custom existant : `mv ~/www/geonature_${GNB}_v${GNV}/frontend/src/custom ~/www/geonature_${GNB}_v${GNV}/frontend/src/custom.save`
    - Créer un nouveau dossier : `mkdir ~/www/geonature_${GNB}_v${GNV}/frontend/src/custom/`
    - Copier l'ancien dossier : `cp -r ~/geonature/frontend/src/custom/* ~/www/geonature_${GNB}_v${GNV}/frontend/src/custom/`
    - Afficher les différences entre les 2 dossiers : `cd ~/www/geonature_${GNB}_v${GNV}/frontend/src ; diff -qr custom.save/ custom/`
- Copier les fichiers statics du backend :
    - `export` avec : `cp -r ~/geonature/backend/static/exports/* ~/www/geonature_${GNB}_v${GNV}/backend/static/exports`
- Copier le favicon avec : `cp ~/geonature/frontend/src/favicon.ico ~/www/geonature_${GNB}_v${GNV}/frontend/src/favicon.ico`
- Copier la lib leaflet-draw (corrigée) : `cp ~/www/geonature_old/frontend/node_modules/leaflet-draw/dist/leaflet.draw.js ~/www/geonature/frontend/node_modules/leaflet-draw/dist/leaflet.draw.js`
- Modifier les liens symboliques :
    - pour la nouvelle version : `cd ~/www; rm -f geonature; ln -s geonature_${GNB}_v${GNV} geonature`
    - pour l'ancienne version :
        - `cd ~/www; rm -f geonature_old; ln -s geonature_v${GNV} geonature_old`
        - ou si l'ancienne version était déjà sur une branche de test : `cd ~/www; rm -f geonature_old; ln -s geonature_${GNB}_v<ancienne-version> geonature_old`
- Vérifier qu'il n'y pas de nécessité de réaliser une migration de la base de GeoNature: https://github.com/PnX-SI/GeoNature/releases

- Exécuter les scripts de mise à jour de la base de données :
  - **ATTENTION** : remettre à `false` le paramètre `drop_apps_db` du fichier `~/www/geonature_${GNB}_v${GNV}/config/settings.ini` si vous l'avez modifié !
  - GeoNature > 2.8.1 :
    - Utiliser (dans le venv activé) la commande : `geonature db status`
    - Utiliser la commande de mise à jour auto sauf mention contraire dans les notes de releases : `geonature db autoupgrade`
  - GeoNature =< 2.8.1 :
    - sur "*web-srv*" :
      - Synchroniser le dossier GeoNature de "web-srv" vers "db-srv" : `rsync -av -e "ssh -p <port-ssh-db>" /home/geonat/www/ geonat@db-<region>-sinp:/home/geonat/www/`
    - sur "*db-srv*" :
      - vous pouvez utiliser une commande du type : `psql -h localhost -U geonatadmin -d geonature2db -f ~/geonature/data/migrations/<mon-fichier-de-mise-a-jour>.sql`
- Réactiver GCC le temps de l'install du venv Python (⇒ compilation Psycopg2) : `sudo chmod o+x /usr/bin/gcc`
- Créer le fichier cron.log du module Export (voir #86) : `mkdir -p /home/${USER}/geonature/var/log/gn_export/ ; touch /home/${USER}/geonature/var/log/gn_export/cron.log`
- Migrer GeoNature : `cd ~/www/geonature/ ; ./install/migration/migration.sh`
- Désactiver à nouveau GCC : `sudo chmod o-x /usr/bin/gcc`
- **ATTENTION** : remettre à `false` le paramètre `drop_apps_db` du fichier `~/www/geonature_${GNB}_v${GNV}/config/settings.ini` si vous l'avez modifié !
- Vérifier le statuts du service GeoNature :
  - GeoNature > 2.8.1 : via SystemD avec `sudo systemctl status geonature`
  - GeoNature =< 2.8.1 : via Supervisord avec `sudo supervisorctl status geonature2`

UsersHub

Toutes étapes suivantes se déroulent sur "*web-srv*" à l'exception des mises à jour de la base sur "*db-srv*" :

- Exporter le nom de la branche dans une variable d'env locale à la session. C'est le nom utilisé pour les fichiers il faut donc y remplacer les éventuels "/" par "-" : `export UHB="<branche>"`
- Indiquer la version de UsersHub sur laquelle se base les développements
  - Pour voir la version, en local dans le dossier du code de UsersHub contenant les développements : `cat VERSION`
  - Récupérer aussi les 7 derniers caractères du SHA1 du dernier commit de la branche à récupérer
  - Exporter la version de UsersHub obtenu avec *cat* en la concaténant avec les 7 premiers caractères du commit séparé par "-" . Ex. : `export UHV="2.2.2-0580563"`
- Vérifier les versions en question : `echo "Branche: ${UHB} ; UsersHub : ${UHV}"`
- Se placer dans le dossier *dwl* de l'utilisateur *geonat* : `cd ~/dwl/`
- Télécharger l'archive : `wget`

`https://github.com/PnX-SI/UsersHub/archive/<branche>.zip -O "usershub_${UHB}_v${UHV}.zip"`
    - Il est aussi possible d'indiquer un SHA1 de commit pour l'URL : `https://github.com/PnX-SI/UsersHub/archive/<sha1>.zip`
- Décompresser l'archive dans le dossier *dwl* de l'utilisateur *geonat* : `unzip usershub_${UHB}_v${UHV}.zip -d ~/dwl/`
- Renommer le dossier au format GeoNature (⇒ uniformité), dans le nom de la branche les "/" sont remplacés par des "-" : `mv UsersHub-${UHB} usershub_${UHB}_v${UHV}`
- Déplacer le dossier au format GeoNature dans *www/* : `mv usershub_${UHB}_v${UHV} ~/www/`
- Se rendre dans le dossier *www* : `cd ~/www/`
- Copier les fichiers de config de l'ancien dossier vers le nouveau :
    - le fichier de paramètres pour les scripts d'installation et Gunicorn : `cp ~/www/usershub/config/settings.ini ~/www/usershub_${UHB}_v${UHV}/config/settings.ini`
    - le fichier de config du backend Python : `cp ~/www/usershub/config/config.py ~/www/usershub_${UHB}_v${UHV}/config/config.py`
- Vérifier l'existence de nouveaux paramètres de config :
    - Dans *config.py* : `diff ~/www/usershub/config/config.py.sample ~/www/usershub_${UHB}_v${UHV}/config/config.py.sample`
    - Dans *settings.ini* : `diff ~/www/usershub/config/settings.ini.sample ~/www/usershub_${UHB}_v${UHV}/config/settings.ini.sample`
- Compléter/modifier les paramètres de config en fonction des besoins de la branches dans les fichiers :
    - *config.py* : `vi ~/www/usershub_${UHB}_v${UHV}/config/config.py`
    - *settings.ini* : `vi ~/www/usershub_${UHB}_v${UHV}/config/settings.ini`
- Modifier les liens symboliques :
    - pour la nouvelle version : `cd ~/www; rm -f usershub; ln -s usershub_${UHB}_v${UHV} usershub`
    - pour l'ancienne version :
        - `cd ~/www; rm -f usershub_old; ln -s usershub_v${UHV} usershub_old`
        - ou si l'ancienne version était déjà sur une branche de test : `cd ~/www; rm -f usershub_old; ln -s usershub_${UHB}_v<ancienne-version> usershub_old`
- La mise à jour de la base de données passe par la commande `geonature` depuis la version 2.9.0, voir cet étape dans la section consacré à GeoNature.
- Consulter les éventuelles notes de version spécifiques décrites au niveau de chaque version : https://github.com/PnX-SI/UsersHub/releases
    - Sur l'instance "*web-srv*" exécuter tous les scripts en lien avec l'interface.

TaxHub

Toutes étapes suivantes se déroulent sur "*web-srv*" à l'exception des mises à jour de la base sur "*db-srv*" :

- Exporter le nom de la branche dans une variable d'env locale à la session. C'est le nom utilisé pour les fichiers il faut donc y remplacer les éventuels "/" par "-" : `export THB="<branche>"`
- Indiquer la version de TaxHub sur laquelle se base les développements
    - Pour voir la version, en local dans le dossier du code de TaxHub contenant les

développements : `cat VERSION`
    - Récupérer aussi les 7 derniers caractères du SHA1 du dernier commit de la branche à récupérer
    - Exporter la version de TaxHub obtenu avec *cat* en la concaténant avec les 7 premiers caractères du commit séparé par "-" . Ex. : `export THV="1.9.4-0580563"`
- Vérifier les versions en question : `echo "Branche: ${THB} ; TaxHub : ${THV}"`
- Se placer dans le dossier *dwl* de l'utilisateur *geonat* : `cd ~/dwl/`
- Télécharger l'archive : `wget https://github.com/PnX-SI/TaxHub/archive/<branche>.zip -O "taxhub_${THB}_v${THV}.zip"`
    - Il est aussi possible d'indiquer un SHA1 de commit pour l'URL : `https://github.com/PnX-SI/TaxHub/archive/<sha1>.zip`
- Décompresser l'archive dans le dossier *dwl* de l'utilisateur *geonat* : `unzip taxhub_${THB}_v${THV}.zip -d ~/dwl/`
- Renommer le dossier au format GeoNature (⇒ uniformité), dans le nom de la branche les "/" sont remplacés par des "-" : `mv TaxHub-${THB} taxhub_${THB}_v${THV}`
- Déplacer le dossier au format GeoNature dans *www/* : `mv taxhub_${THB}_v${THV} ~/www/`
- Se rendre dans le dossier *www* : `cd ~/www/`
- Copier les fichiers de config de l'ancien dossier vers le nouveau :
    - le fichier de paramètres pour les scripts d'installation et Gunicorn : `cp ~/www/taxhub/settings.ini ~/www/taxhub_${THB}_v${THV}/settings.ini`
    - le fichier de config du backend Python : `cp ~/www/taxhub/apptax/config.py ~/www/taxhub_${THB}_v${THV}/apptax/config.py`
        - Pour TaxHub < 1.9.2 : `cp ~/www/taxhub/config.py ~/www/taxhub_${THB}_v${THV}/config.py`
    - le fichier de config du frontend Javascript : `cp ~/www/taxhub/static/app/constants.js ~/www/taxhub_${THB}_v${THV}/static/app/constants.js`
- Transférer les médias uploadés dans l'ancienne version vers la nouvelle : `cp -aR ~/www/taxhub/static/medias/ ~/www/taxhub_${THB}_v${THV}/static/medias/`
    - Vérifier les droits des dossiers et fichiers copiés
- Vérifier l'existence de nouveaux paramètres de config :
    - Dans *config.py* : `diff ~/www/taxhub/apptax/config.py.sample ~/www/taxhub_${THB}_v${THV}/apptax/config.py.sample`
        - Pour TaxHub < 1.9.2 : `diff ~/www/taxhub/config.py.sample ~/www/taxhub_${THB}_v${THV}/config.py.sample`
    - Dans *settings.ini* : `diff ~/www/taxhub/settings.ini.sample ~/www/taxhub_${THB}_v${THV}/settings.ini.sample`
    - Dans *constants.js* : `diff ~/www/taxhub/static/app/constants.js.sample ~/www/taxhub_${THB}_v${THV}/static/app/constants.js.sample`
- Compléter/modifier les paramètres de config en fonction des besoins de la branches dans les fichiers :
    - *config.py* : `vi ~/www/taxhub_${THB}_v${THV}/config.py`
    - *settings.ini* : `vi ~/www/taxhub_${THB}_v${THV}/settings.ini`
    - *constants.js* : `vi ~/www/taxhub_${THB}_v${THV}/static/app/constants.js`
- Modifier les liens symboliques :
    - pour la nouvelle version : `cd ~/www; rm -f taxhub; ln -s taxhub_${THB}_v${THV} taxhub`
    - pour l'ancienne version :

        - `cd ~/www; rm -f taxhub_old; ln -s taxhub_v${THV} taxhub_old`
        - ou si l'ancienne version était déjà sur une branche de test : `cd ~/www; rm -f taxhub_old; ln -s taxhub_${THB}_v<ancienne-version> taxhub_old`
- La mise à jour de la base de données passe par la commande `geonature` depuis la version 2.9.0, voir cet étape dans la section consacré à GeoNature.
- Consulter les éventuelles notes de version spécifiques décrites au niveau de chaque version : https://github.com/PnX-SI/TaxHub/releases
    - Sur l'instance "*web-srv*" exécuter tous les scripts en lien avec l'interface.

Atlas

Voir la page dédié à la mise à jour de GeoNature Atlas.

Corrections des fichiers de GeoNature spécifique à notre installation

Surcouchage des fichiers Systemd de GeoNature (GeoNature > v2.8.1)

Passage à Systemd :

- Copier le fichier `install/assets/geonature.service` dans `/etc/systemd/system/` : `sudo cp ~/www/geonature_${GNB}_v${GNV}/install/assets/geonature.service /etc/systemd/system`
- Éditer `/etc/systemd/system/geonature.service` et remplacer les variables `${USER}` (votre utilisateur linux courant) et `${BASE_DIR}` (chemin absolu du répertoire de GeoNature) par les valeurs appropriées
- Créer un fichier `environ` à la racine de votre dossier GeoNature, et indiquer les variables d'environnement avec les valeurs modifiées :

```
GUNICORN_NUM_WORKERS=9
GUNICORN_TIMEOUT=300
```

- Supprimer les paramètres liés à Gunicorn présent dans le fichier `config/settings.ini` de GeoNature.
- Lancer la commande `sudo systemctl daemon-reload`
- Pour démarrer GeoNature : `sudo systemctl start geonature`
- Pour lancer GeoNature automatiquement au démarrage du serveur : `sudo systemctl enable geonature`

Surcoucher le service Systemd de GeoNature :

- Afin d'éviter que les modifications effectuées dans le fichier */lib/systemd/system/geonature.service* soient écrasées à chaque mise à jour de GeoNature, vous devez ajouter un fichier qui écrasera les valeurs par défaut.
- Pour créer automatiquement l'arborescence de dossier et le fichier nécessaire, utiliser la commande suivante : `systemctl edit geonature`
- La commande précédente ouvre l'éditeur par défaut du système, vous pouvez ajouter le contenu suivant et sortir de l'édition du fichier en sauvegardant :

```
[Unit]
StartLimitIntervalSec=60min
StartLimitBurst=25

[Service]
ExecStart=
ExecStart=/home/geonat/www/geonature/backend/venv/bin/gunicorn
geonature:create_app() \
                --statsd-host "localhost:8125" \
                --statsd-prefix "geonature"  \
                --name "${GUNICORN_PROC_NAME}" --workers
"${GUNICORN_NUM_WORKERS}" \
                --bind "${GUNICORN_HOST}:${GUNICORN_PORT}" --
timeout="${GUNICORN_TIMEOUT}"
Restart=on-failure
RestartSec=2min
StandardOutput=append:/var/log/geonature/geonature.log
```

  - **Note** : la première ligne *ExecStart*= vide permet de réinitialiser la commande de lancement
- Les modifications devraient être présente dans le fichier suivant : `vi /etc/systemd/system/geonature.service.d/override.conf`
- Lancer la prise en compte des modifications qui vérifiera une éventuelle erreur : `systemctl daemon-reload`
- Relancer le service si nécessaire : `systemctl restart geonature`

Surcouchage des fichiers Systemd de TaxHub (TaxHub > v1.9.0)

Surcoucher le service Systemd de TaxHub :

- Afin d'éviter que les modifications effectuées dans le fichier */lib/systemd/system/taxhub.service* soient écrasées à chaque mise à jour de TaxHub, vous devez ajouter un fichier qui écrasera les valeurs par défaut.
- Pour créer automatiquement l'arborescence de dossier et le fichier nécessaire, utiliser la commande suivante : `systemctl edit taxhub`
- La commande précédente ouvre l'éditeur par défaut du système, vous pouvez ajouter le contenu suivant et sortir de l'édition du fichier en sauvegardant :

```
[Unit]
StartLimitIntervalSec=60min
StartLimitBurst=25

[Service]
ExecStart=
ExecStart=/home/geonat/www/taxhub/venv/bin/gunicorn
apptax.app:create_app() \
                --statsd-host "localhost:8125" \
                --statsd-prefix "taxhub"  \
```

```
                --name "${GUNICORN_PROC_NAME}" --workers
"${GUNICORN_NUM_WORKERS}" \
                --bind "${GUNICORN_HOST}:${GUNICORN_PORT}" --
timeout="${GUNICORN_TIMEOUT}"
Restart=on-failure
RestartSec=2min
StandardOutput=append:/var/log/taxhub/taxhub.log
```

  - **Note** : la première ligne *ExecStart*= vide permet de réinitialiser la commande de lancement
- Les modifications devraient être présente dans le fichier suivant : `vi /etc/systemd/system/taxhub.service.d/override.conf`
- Lancer la prise en compte des modifications qui vérifiera une éventuelle erreur : `systemctl daemon-reload`
- Relancer le service si nécessaire : `systemctl restart taxhub`

Surcouchage des fichiers Systemd de UsersHub (UsersHub > v2.1.3)

Surcoucher le service Systemd de UsersHub :

- Afin d'éviter que les modifications effectuées dans le fichier */lib/systemd/system/usershub.service* soient écrasées à chaque mise à jour de UsersHub, vous devez ajouter un fichier qui écrasera les valeurs par défaut.
- Pour créer automatiquement l'arborescence de dossier et le fichier nécessaire, utiliser la commande suivante : `systemctl edit usershub`
- La commande précédente ouvre l'éditeur par défaut du système, vous pouvez ajouter le contenu suivant et sortir de l'édition du fichier en sauvegardant :

```
[Unit]
StartLimitIntervalSec=60min
StartLimitBurst=25

[Service]
ExecStart=
ExecStart=/home/geonat/www/usershub/venv/bin/gunicorn
app.app:create_app() \
                --statsd-host "localhost:8125" \
                --statsd-prefix "usershub"  \
                --name "${GUNICORN_PROC_NAME}" --workers
"${GUNICORN_NUM_WORKERS}" \
                --bind "${GUNICORN_HOST}:${GUNICORN_PORT}" --
timeout="${GUNICORN_TIMEOUT}"
Restart=on-failure
RestartSec=2min
StandardOutput=append:/var/log/usershub/usershub.log
```

  - **Note** : la première ligne *ExecStart*= vide permet de réinitialiser la commande de lancement
- Les modifications devraient être présente dans le fichier suivant : `vi /etc/systemd/system/usershub.service.d/override.conf`

- Lancer la prise en compte des modifications qui vérifiera une éventuelle erreur : `systemctl daemon-reload`
- Relancer le service si nécessaire : `systemctl restart usershub`

Fichier "backend/gunicorn_start.sh" (GeoNature =< v2.8.1)

Ajout des options `--statsd-*` :

```
exec gunicorn  wsgi:app \
    --statsd-host=localhost:8125 \
    --statsd-prefix=geonature  \
    --timeout ${gun_timeout} \
    --error-log $APP_DIR/var/log/gn_errors.log \
    --pid="${app_name}.pid" \
    -w "${gun_num_workers}" \
    -b "${gun_host}:${gun_port}" \
    -n "${app_name}"
```

Mises à jour

GeoNature

v2.7.5 --> 2.10.0.dev0 (feat/sinp)

- Sur "*web-srv*" :
  - `sudo supervisorctl stop geonature2`
  - `sudo rm /etc/supervisor/conf.d/geonature-service.conf`
  - Corriger la directive proxy_passe de la conf GeoNature de Nginx :
    - `sudo vi /etc/nginx/sites-available/geonature.conf`
      - Remplacer `proxy_pass http://127.0.0.1:8000/;` par `proxy_pass http://127.0.0.1:8000/api/;`
    - Recharger la config : `nginx-reload`
  - Passage à Systemd et surcouchage du service, voir la section dédiée dans la mise à jour de GeoNature.
  - Modifier le fichier `config/geonature_config.toml` en :
    - ajoutant (changer le chemin en local):

      ```
      [ALEMBIC]
      VERSION_LOCATIONS =
      "/home/geonat/www/usershub/app/migrations/versions/"
      ```

    - remplaçant les `id_type` par des `type_code` dans le paramètre AREA_FILTERS. Par exemple, l'id 25 devient "COM".
    - remplacer les `id_type` par des `type_code` dans le paramètre AREA_TYPES. Ex. : `AREA_TYPES = ["COM", "DEP", "SINP"]`

    - `cd install/`
    - `./01_install_backend.sh`
    - `source backend/venv/bin/activate`
    - `geonature db stamp f06cc80cc8ba`
    - `geonature db stamp 0dfdbfbccd63`
    - `geonature db stamp 3fdaa1805575`
    - `geonature db stamp 586613e2faeb`
    - `geonature db stamp 7d6e98441e4c`
    - `geonature db stamp ede150d9afd9`
- Sur "*db-srv*" :
    - `psql -h localhost -U geonatadmin -d geonature2db -c "DELETE FROM gn_sensitivity.cor_sensitivity_criteria WHERE id_criteria IS NULL;"`
- Sur "*web-srv*" :
    - `geonature db status` chercher la révision sur la brance `[geonature]` qui précède `3b2f3de760dc access request`.
    - `geonature db upgrade ca052245c6ec -x recompute-sensitivity=false -x restore-diffusion-level=false`
    - `geonature db stamp c9854947fa23` # data-blurring
    - `geonature db upgrade geonature@head`
    - Vérifier l'état de la base : `geonature db status`
    - Mise à jour des branches :
        - `geonature db upgrade habitats_inpn_data@head`
        - `geonature db upgrade nomenclatures@head`
        - `geonature db upgrade taxonomie@head`
    - `cd ~/geonature/frontend/`
    - `nvm install`
    - `nvm use`
    - `cd ~/geonature/install/`
    - `./04_install_frontend.sh`
    - `cd ~/geonature/frontend`
    - `npm i -D @angular-builders/custom-webpack`
    - `npm run build`
    - Faire la mise à jour de TaxRef v15 via TaxHub.
        - Estampiller la base avec la révision correspondant à la mise à jour TaxRef v15 : `geonature db stamp ad35a3bf85c2`
        - Finaliser la mise à jour de la branche *taxonomie_inpn_data* avec : `geonature db upgrade taxonomie_inpn_data@head`
    - `sudo systemctl restart geonature`
    - `sudo aptitude remove supervisor`
- Sur "*db-srv*" :
    - Procéder à la mise à jour des données CEN-PACA, CBNA, CBNMED.
    - `psql -h localhost -U geonatadmin -d geonature2db -f ~/data/db-geonature/data/sql/04_*`
    - `psql -h localhost -U geonatadmin -d geonature2db -f ~/data/db-geonature/data/sql/01_fix_v_synthese_for_export.sql`
    - `psql -h localhost -U geonatadmin -d geonature2db -f ~/data/cenpaca/data/sql/fix/006_*`

v2.6.3.dev0 --> 2.7.5 (feat/sinp)

- Sur "*web-srv*" :
  - Suivre les étapes de l'installation de GeoNature sur une branche hors releases
  - `mv ~/geonature/frontend/src/custom/custom.scss ~/geonature/frontend/src/assets/custom.css`
- Sur "*db-srv*" :
  - `psql -h localhost -U geonatadmin -d geonature2db -f "data/migrations/2.6.2to2.7.0.sql"`
  - En local : `rsync -av ./ geonat@db-paca-sinp:~/data/db-geonature/ --dry-run` supprimer –dry-run si tout est ok
  - `psql -h localhost -U geonatadmin -d geonature2db -f ~/data/db-geonature/data/sql/01_fix_v_synthese_for_export.sql`
  - `psql -h localhost -U geonatadmin -d geonature2db -f "data/migrations/2.7.1to2.7.2.sql"`
  - Mise à jour de UsersHub en version v2.2.2

Test migration en local : v2.6.3.dev0 --> 2.10.0.dev0 (feat/sinp)

- Page détaillée de la mise à jour
- Notes sur le test de migration en local :
  - Commencer par la mise à jour de Taxhub en suivant les indications : de v1.7.4.dev0 vers v1.9.3
    - ```
      psql -h localhost -U geonatadmin -d gn2_dev_sinp -f "data/update1.7.3to1.8.0.sql"
      ```
  - Mise à jour vers TaxRef v13 : doc
  - Mise à jour vers TaxRef v14 : doc
    - Nécessite de mettre à jour dans "gn_synthese.synthese" les cd_nom supprimés listés dans *liste_cd_nom_disparus_bib_noms.csv* et dans *liste_cd_nom_disparus_synthese.csv*. Les requêtes SQL doivent être placées dans le fichier *2.1_taxref_changes_corrections_pre_detections.sql*.
    - Pour les cd_nom sans remplaçant supprimer les observations liés s'il n'est pas possible de trouver le bon cd_nom pour les obs en question.
    - Dans le fichiers *liste_changements.csv*, rechercher dans la colonne "*action*" les entrées `Loose attributes and medium now attach to …` et "Conflict …" pour lesquels il peut y avoir une action à mener si des attributs (*att_nb*) ou des media existent (*media_nb*).
  - Mise à jour de la base vers v2.7.0 :

    ```
    psql -h localhost -U geonatadmin -d gn2_dev_sinp -f "data/migrations/2.6.2to2.7.0.sql"
    ```

    - Mettre à jour la vue "gn_synthese.v_synthese_for_export" suite à la migration : SINP AURA, SINP PACA.
      - ```
        psql -h localhost -U geonatadmin -d gn2_dev_sinp -f "data/sql/01_fix_v_synthese_for_export.sql"
        ```
  - Changement du fichier *custom.scss* :

        - Modifier le chemin vers le fichier utilisé dans le fichier `frontend/src/assets/custom.css`
        - Le fichier est en CSS et plus en SCSS le modifier en conséquence
        - Changer son emplacement sur les dépôts sinp-paca-srv et sinp-aura-srv.
    - Mise à jour de la base vers v2.7.2 :

```
psql -h localhost -U geonatadmin -d gn2_dev_sinp -f "data/migrations/2.7.1to2.7.2.sql"
```

v2.6.1 --> 2.6.3.dev0

- Réinitialisation de la base de données

v2.6.0.rc.1 --> 2.6.1

- Sur "*db-srv*" :
    - `cd ~/geonature`
    - `mkdir tmp`
    - `vi tmp/2.6.0-rc.1to2.6.0.sql` : placer dedans toutes les lignes du script de migration situé après la ligne 471 indiquant "---- AFTER 2.6.0.rc.1". Penser à rajouter un `BEGIN;` au début du fichier.
    - `wget https://raw.githubusercontent.com/PnX-SI/Nomenclature-api-module/master/data/update1.3.4to1.3.5.sql -O tmp/nomenclature-api-module_update1.3.4to1.3.5.sql`
    - `wget https://raw.githubusercontent.com/PnX-SI/Nomenclature-api-module/master/data/update1.3.4to1.3.5.sql -O tmp/nomenclature-api-module_update1.3.5to1.3.6.sql`
    - `psql -h localhost -U geonatadmin -d geonature2db -f ./tmp/2.6.0-rc.1to2.6.0.sql`
    - `sudo -u postgres psql -d geonature2db -c "UPDATE pg_attribute SET atttypmod = 5004 WHERE attrelid = 'gn_synthese.synthese'::regclass AND attname = 'reference_biblio';"`
    - `psql -h localhost -U geonatadmin -d geonature2db -f ./tmp/nomenclature-api-module_update1.3.4to1.3.5.sql`
    - `psql -h localhost -U geonatadmin -d geonature2db -f ./tmp/nomenclature-api-module_update1.3.5to1.3.6.sql`
    - `psql -h localhost -U geonatadmin -d geonature2db -f ./data/migrations/2.6.0to2.6.1.sql`
    - `psql -h localhost -U geonatadmin -d geonature2db -f ./data/migrations/access_request.sql`
- Sur "*web-srv*" :
    - Rien de particulier

v2.5.5 --> 2.6.0.rc.1

- Sur "*db-srv*" :
  - `psql -h localhost -U geonatadmin -d geonature2db -f ~/geonature/data/migrations/2.5.5to2.6.0_maj_jpm_2021-01-14.sql` : ce fichier *2.5.5to2.6.0_maj_jpm_2021-01-14.sql* contient toutes les lignes du fichier *2.5.5to2.6.0.sql* du début du fichier jusqu'à la ligne de commentaire `—- AFTER 2.5.6.rc.2`.
- Sur "*web-srv*" :
  - Copier le fichier *response.py* de la lib python *utils_flask_sqla* qui contient une modification (voir https://github.com/PnX-SI/Utils-Flask-SQLAlchemy/issues/15) dans la nouvelle version : `cp ~/geonature_old/backend/venv/lib/python3.7/site-packages/utils_flask_sqla/response.py ~/geonature/backend/venv/lib/python3.7/site-packages/utils_flask_sqla/response.py`

v2.3.2 --> 2.5.3

- Sur "*db-srv*" :
  - `cd ~/www/geonature/data/migrations ; wget https://raw.githubusercontent.com/PnX-SI/Nomenclature-api-module/master/data/update1.3.2to1.3.3.sql -O nomenclature-api-module_update1.3.2to1.3.3.sql`
  - `psql -h localhost -U geonatadmin -d geonature2db -f ~/geonature/data/migrations/nomenclature-api-module_update1.3.2to1.3.3.sql`
  - `psql -h localhost -U geonatadmin -d geonature2db -f ~/geonature/data/migrations/2.3.2to2.4.0.sql`
  - **NOTE** : La requête `DELETE FROM gn_synthese.cor_area_synthese cas USING gn_synthese.synthese s, ref_geo.l_areas a WHERE cas.id_synthese = s.id_synthese AND a.id_area = cas.id_area AND public.ST_TOUCHES(s.the_geom_local,a.geom);` **n'a pas** été exécuté... Voir ticket #719.
  - `psql -h localhost -U geonatadmin -d geonature2db -f ~/geonature/data/migrations/2.4.0to2.4.1.sql`
  - `mkdir ~/www/geonature/var/log/`
  - `cd ~/www/geonature/data/migrations/ ; ./2.4.0to2.4.1_insert_departments.sh`
  - `cd ~/www/geonature/data/migrations ; wget https://raw.githubusercontent.com/PnX-SI/Nomenclature-api-module/master/data/update1.3.3to1.3.4.sql -O nomenclature-api-module_update1.3.3to1.3.4.sql`
  - `psql -h localhost -U geonatadmin -d geonature2db -f ~/geonature/data/migrations/nomenclature-api-module_update1.3.3to1.3.4.sql`
  - A à l'aide de DBeaver copier/coller dans un fichier le source de la vue `gn_exports.v_synthese_sinp` puis supprimer cette vue.
  - `psql -h localhost -U geonatadmin -d geonature2db -f ~/geonature/data/migrations/2.4.1to2.5.0.sql`
  - Recréer la vue `gn_exports.v_synthese_sinp` en effectuant au préalable des

corrections dedans : supprimer le champ id_nomenclature_obs_technique et renommer le champ id_nomenclature_obs_meth ⇒ id_nomenclature_obs_technique, remplacer le label "obsMeth" par "obsTech".
    - `psql -h localhost -U geonatadmin -d geonature2db -f ~/geonature/data/migrations/2.5.1to2.5.2.sql`
    - `psql -h localhost -U geonatadmin -d geonature2db -f ~/geonature/data/migrations/2.5.2to2.5.3.sql`
- Sur "*web-srv*" :
    - `sudo aptitude install libcairo2 libpango-1.0-0 libpangocairo-1.0-0 libgdk-pixbuf2.0-0 libffi-dev shared-mime-info`
    - `mkdir -p /home/${USER}/geonature/var/log/gn_export/ ; touch /home/${USER}/geonature/var/log/gn_export/cron.log`
    - `sudo chmod o+x /usr/bin/gcc` : réactiver GCC le temps de l'install du venv Python (⇒ compilation Psycopg2)
    - `cd ~/www/geonature/ ; ./install/migration/migration.sh`
    - `sudo chmod o-x /usr/bin/gcc`
    - `sudo supervisorctl status geonature2`

v2.2.1 --> 2.3.2

- Installation du module *OccHab* : `geonature install_gn_module /home/geonat/www/geonature/contrib/gn_module_occhab /occhab`
- Modifier le nom et l'icône du menu `occhab` en suivant la procédure.

GeoNature - Module Export

Voir la page d'installation et mise à jour du module Export.

UsersHub

v2.1.3 --> v2.2.2

- Sur "*web-srv*" :
    - Suivre les étapes de l'installation de UsersHub sur une branche hors releases
    - `sudo supervisorctl stop usershub2`
    - `sudo rm /etc/supervisor/conf.d/usershub-service.conf`
    - Passage à Systemd et surcouchage du service, voir la section dédiée dans la mise à jour de GeoNature.
    - `sudo aptitude install python3-venv`
    - Supprimer dans le fichier `config/config.py` le paramètre `URLS_COR`, il n'est plus utilisé.
    - Lancer la mise à jour de l'application : `./install_app.sh`
    - Corriger le chemin vers UsersHub dans le fichier : `vi /etc/systemd/system/usershub.service`
- Sur "*db-srv*" :
    - La mise à jour de la base doit passer par la commande `geonature` via Alembic. Il est donc nécessaire de mettre à jour GeoNature en v2.10.0.dev0.

        - Généralement utilisé : `geonature db upgrade usershub@head`

v2.1.0 --> v2.1.3

- Sur "*db-srv*" :
    - `psql -h localhost -U geonatadmin -d geonature2db -f ~/usershub/data/update_2.1.0to2.1.1.sql`
    - `psql -h localhost -U admin -d geonature2db -c "UPDATE pg_attribute SET atttypmod = 500+4 WHERE attrelid = 'utilisateurs.bib_organismes'::regclass AND attname = 'nom_organisme';"`
    - `psql -h localhost -U geonatadmin -d geonature2db -f ~/usershub/data/update_2.1.2to2.1.3.sql`
- Sur "*web-srv*" :
    - `sudo aptitude install python3-pip`
    - `sudo pip3 install virtualenv==20.0.1`
    - `sudo chmod o+x /usr/bin/gcc` : réactiver GCC le temps de l'install du venv Python (⇒ compilation Psycopg2)
    - `cd ~/www/usershub/ ; rm -fR venv ; ./install_app.sh`
    - `sudo chmod o-x /usr/bin/gcc`
    - `sudo supervisorctl status usershub`

TaxHub

v1.7.3 --> v1.9.4 (feat/sinp)

- Sur "*web-srv*" :
    - Suivre les étapes de l'installation de TaxHub sur une branche hors releases
    - `sudo supervisorctl stop taxhub`
    - `sudo rm /etc/supervisor/conf.d/taxhub-service.conf`
    - Passage à Systemd et surcouchage du service, voir la section dédiée dans la mise à jour de GeoNature.
    - `sudo aptitude install python3-venv`
    - `cd ~/taxhub`
    - `mv config.py apptax/`
    - Modifier le fichier `apptax/config.py` ainsi :
        - Supprimer les lignes :

```
# File
import os
BASE_DIR = os.path.abspath(os.path.dirname(__file__))
```

        - Remplacer le paramètre `UPLOAD_FOLDER = 'static/medias'` par `UPLOAD_FOLDER = 'medias'`
    - Lancer la mise à jour de l'application : `./install_app.sh`
    - Corriger le chemin vers TaxHub dans le fichier : `vi /etc/systemd/system/taxhub.service`

    - Passage à TaxRef v15 :
        - `cd ~/www/taxhub/`
        - `source venv/bin/activate`
        - `export FLASK_APP=apptax/app.py`
        - `flask taxref_migration import_taxref_v15`
        - `wget https://raw.githubusercontent.com/cbn-alpin/sinp-paca-data/master/taxref/v15/sql/2.1_taxref_changes_corrections_pre_detections.sql -O apptax/migrations/data/migrate_taxref_version/2.1_taxref_changes_corrections_pre_detections.sql`
        - `wget https://raw.githubusercontent.com/cbn-alpin/sinp-paca-data/master/taxref/v15/sql/2.2_taxref_changes_corrections_post_detections.sql -O apptax/migrations/data/migrate_taxref_version/2.2_taxref_changes_corrections_post_detections.sql`
        - `flask taxref_migration apply_changes --script_predetection /home/geonat/www/taxhub/apptax/migrations/data/migrate_taxref_version/2.1_taxref_changes_corrections_pre_detections.sql --script_postdetection /home/geonat/www/taxhub/apptax/migrations/data/migrate_taxref_version/2.2_taxref_changes_corrections_post_detections.sql`
        - Retourner dans GeoNature pour estampiller la base avec la revision correspondant à la mise à jour TaxRef v15 : `geonature db stamp ad35a3bf85c2`
- Sur "*db-srv*" :
    - `cd ~/www/taxhub_feat-sinp_v1.9.4-0580563/`
    - `sudo -u postgres -s psql -d geonature2db -c 'CREATE EXTENSION IF NOT EXISTS "unaccent";'`
    - `psql -h localhost -U geonatadmin -d geonature2db -f "data/update1.7.3to1.8.0.sql"`
    - `cd data/scripts/update_taxref/`
    - `sudo ls`
    - `./import_taxref_v13_data.sh`
    - `wget https://raw.githubusercontent.com/cbn-alpin/sinp-paca-data/master/taxref/v13/data/sql/2.1_taxref_changes_corrections_pre_detections.sql -O scripts/2.1_taxref_changes_corrections_pre_detections.sql`
    - `wget https://raw.githubusercontent.com/cbn-alpin/sinp-paca-data/master/taxref/v13/data/sql/2.2_taxref_changes_corrections_post_detections.sql -O scripts/2.2_taxref_changes_corrections_post_detections.sql`
    - `wget https://raw.githubusercontent.com/cbn-alpin/sinp-paca-data/master/taxref/v13/data/sql/4.3_restore_local_constraints.sql -O scripts/4.3_restore_local_constraints.sql`
    - `./apply_changes.sh 13`
    - `./clean_db.sh`
    - `sudo ls`
    - `./import_taxref_v14.sh`
    - `wget`

    - `https://raw.githubusercontent.com/cbn-alpin/sinp-paca-data/master/taxref/v14/data/sql/2.1_taxref_changes_corrections_pre_detections.sql -O scripts/2.1_taxref_changes_corrections_pre_detections.sql`
    - `wget https://raw.githubusercontent.com/cbn-alpin/sinp-paca-data/master/taxref/v14/data/sql/2.2_taxref_changes_corrections_post_detections.sql -O scripts/2.2_taxref_changes_corrections_post_detections.sql`
    - `wget https://raw.githubusercontent.com/cbn-alpin/sinp-paca-data/master/taxref/v14/data/sql/4.3_restore_local_constraints.sql -O scripts/4.3_restore_local_constraints.sql`
    - `./apply_changes.sh 14`
    - `./clean_db.sh`
    - Mise à jour de GeoNature en version v2.7.5 puis 2.10.0.dev0
    - La mise à jour de la base doit passer par la commande `geonature` via Alembic. Il est donc nécessaire de mettre à jour GeoNature en v2.10.0.dev0.
        - Généralement utilisé : `geonature db upgrade taxonomie@head`

v1.6.3 --> v1.7.3

- Sur "*db-srv*" :
    - `sudo psql -h localhost -U geonatadmin -d geonature2db -f ~/taxhub/data/update1.6.3to1.6.4.sql`
    - `vi ~/www/taxhub/static/app/constants.js` ⇒ supprimer la ligne contenant *id_application*.
    - `sudo psql -h localhost -U geonatadmin -d geonature2db -f ~/taxhub/data/update1.6.5to1.7.0.sql`
    - `sudo psql -h localhost -U geonatadmin -d geonature2db -f ~/taxhub/data/update1.7.1to1.7.2.sql`
- Sur "*web-srv*" :
    - `vi ~/www/taxhub/static/app/constants.js` ⇒ supprimer la ligne contenant *id_application*.
    - `sudo chmod o+x /usr/bin/gcc` : réactiver GCC le temps de l'install du venv Python (⇒ compilation Psycopg2)
    - `cd ~/www/taxhub/ ; rm -fR venv ; ./install_app.sh`
    - `sudo chmod o-x /usr/bin/gcc`
    - `sudo supervisorctl status taxhub`

GeoNature Atlas

- Voir Mise à jour de GeoNature Altas

From:
http://sinp-wiki.cbn-alpin.fr/ - **CBNA SINP**

Permanent link:
http://sinp-wiki.cbn-alpin.fr/serveurs/installation/web-srv/geonature-mise-a-jour

Last update: **2024/02/09 15:17**