

Nanterre

Initialisation

- Définir le fichier setting

```
export ZOO_SETTINGS_FILE=`pwd`/local_settings.py
```

- Créer les tables

```
./manage.py migrate
```

- Charger les schémas

```
./manage.py loaddata rsu.json
```

- Créer une base temporaire contenant les dumps¹ du RSUv1 et de SWARM (IdP OAuth2 du RSUv1):

```
createdb rsuv1  
psql rsuv1 <nrsu.sql  
psql rsuv1 <swarm_nanterre.sql
```

- Charger les dumps dans zoo et produire une fixture pour créer les utilisateurs dans authentic

```
./manage.py rsu-load-dump "dbname=nanterre_rsu" authentic_users.json
```

- Charger les utilisateurs dans authentic

```
authentic2-multitenant-manage tenant_command loaddata -d connexion-moncompte.nanterre.fr authentic_users.json
```

Attention à ce que le fichier authentic_users.json soit dans un répertoire accessible à l'utilisateur authentic-multitenant

1

<https://files.entrouvert.org/Clients%20et%20contacts/3%20-%20Clients%20en%20cours/Nanterre/inputs/dump%20du%2020170210/>
dernier dump de la preprod (je crois), faut certainement redemander un dump récent

Dédoublonnage

Mise en place

- Définir un seuil de dédoublonnage, disons 80%
- Initialiser la base de dédoublonnage

```
./manage.py rsu-duplicates find --limit 0.8
```

- Mettre en place une mise à jour journalière à partir des derniers individus créés les deux derniers jours:

```
./manage.py rsu-duplicates find --limit 0.8 --days 2
```

À lancer dans un cron régulier

Gestion

- Vérifier les doublons

```
./manage.py rsu-duplicates list
```

- Voir les faux-positifs

```
./manage.py rsu-duplicates list --false-positive
```

- Voir les doublons dédupliqués

```
./manage.py rsu-duplicates list --dedup
```

- Supprimer les doublons en dessous d'un seuil (parce qu'on est descendu trop bas en testant par exemple)

```
./manage.py rsu-duplicates delete --limit 0.8
```