Pour plus de contenu, connectez-vous !

Création de l'instance Odoo

Au plus tôt

1) choix du nom pour leur odoo

ex. erp.coopilote.com, mon-petit-odoo-prefere.coopilote.com, je-kiffe-odoo.coopilote.com, etc...

2) Ajout dans leur DNS `coopilote.com`:

```
je-kiffe-odoo IN CNAME cname-coopilote-erp.grap.coop.
je-kiffe-odoo-preproduction IN CNAME cname-coopilote-erp-preproduction.grap.coop.
je-kiffe-odoo-test IN CNAME cname-coopilote-erp-test.grap.coop.
```

3) Ajouter dans le DNS de `grap.coop`

```
cname-coopilote-erp 300 IN A 51.15.24.177
cname-coopilote-erp-preproduction 3600 IN A 51.254.44.22
cname-coopilote-erp-test 3600 IN A 94.23.17.188
```

4) Déployer leur environnement

```
ansible-playbook odoo-vroum-vroum
```

Déploiement Odoo

Pour serveur-test

Config

Dans ansible-grap/config/odoo/cfg_odoo__grp-test-03.yml

> xxxxx = coopalim par ex

> grp-production-08 = dépend du serveur de prod futur

> xY → s'assurer que les ports sont pas pris → x a incrémenter et Y dernier chiffre de version ?

```
-
name: odoo16_xxxxx
odoo_version: 16
odoo_python_version: python3.10
backup_folder: /backup/grp-production-08/
odoo_restore_and_update: false
odoo_code_url: https://gitlab.com/grap-rhone-alpes/grap-odoo-env.git
odoo_database_prefix: xxxxx
odoo_url: odoo-test.xxxxx.com
odoo_xmlrpc_port: 80xY
odoo_longpolling_port: 81xY
odoo_limit_memory_hard: 1610612736 # 1.5 GB
odoo_limit_memory_soft: 1073741824 # 1.0 GB
odoo_limit_request: 8192
odoo_limit_time_cpu: 3600 # 1 hour
odoo_limit_time_real: 7200 # 2 hours
odoo_workers: 4
odoo_max_cron_threads: 1

 eshops: []

odoo_main_folder: /opt/odoo16_xxxxx/
odoo_filestore_folder: /opt/odoo16_xxxxx/filestore/
odoo_code_folder: /opt/odoo16_xxxxx/code/
odoo_log_folder: /var/log/odoo16_xxxxx/
```

Dans hosts.ini : dé-commenter le serveur de test

Préinstallation

Dans odoo_envs/grap-odoo-env-16.0 Créer une branche à partir de la dernière branche en prod'

```
git checkout –b 16.0-COOPALIM-First-install
Git push origin 16.0-COOPALIM-First-install

time ansible-playbook --limit=grp-test-03@grap.coop ./playbook/p_odoo_install.yml --extra-vars
"{'odoo_settings': [{'name': 'odoo16_xxxxx', 'branch': '16.0-COOPALIM-First-install'},]}"
```

Création de la base

Sur le serveur de test

```
cd /opt/odoo16_coopalim/code

sudo su odoo16_coopalim -c './env/bin/click-odoo-initdb -n coopalim_test_2026_04_09 --no-demo
--no-cache --config ./odoo.cfg'
```

Si on s’est pas merdé, on doit avoir un beau Odoo neuf sur odoo-test.xxxx.com. Connexion avec admin / admin

Aller extraire les modules installés

Sur instance de la même version, extraire la liste des modules installés.
Trafiquer avec Calc et CONCAT pour en faire une liste avec des guillemets et virgules

Dans grap-odoo-env-16.0/scripts/
Soit créer soit rajouter dans le script de la branche qui sera choisie

```
import tools

_logger = tools._logger

@tools.restrict(databases=["xxxxx"])
def coopalim_first_install(env):
    tools.install_modules(
        env,
        [
            # ------------------------------------
            # Export MACHE PRE-PROD 09 avril 2026
            # ------------------------------------
            "sale_management",
            "account",
```

```
            liste à mettre
            "phone_validation"
        ],
    )

def main(env):
    tools.update_module_list(env)

    # COOPALIM FIRST INSTALL
    coopalim_first_install(env)

    tools.install_modules(
        env,
        [
            # ----------------------------
            # Placeholder for Quentin
            # ----------------------------
            # Placeholder for Sylvain
        ],
    )
    tools.uninstall_modules(
        env,
        [
            # ----------------------------
            # Placeholder for Quentin
            # ----------------------------
            # Placeholder for Sylvain
        ],
    )
    tools.update_translations(env)
```

Pousser cette modif sur 16.0-COOPALIM-First-install

```
git add -A
git commit –m « add all modules first install »
git push origin 16.0-COOPALIM-First-install
```

Puis lance

```
time ansible-playbook --limit=grp-test-03@grap.coop ./playbook/p_odoo_update_all.yml --extra-vars "{'odoo_settings': [{'name': 'odoo16_coopalim', 'branch': '16.0-COOPALIM-First-install'},]}"
```

> TODO : rajouter les modifs dans les restaurations quotidienne

Pour serveur de pre-prod

Même process, sauf que la branche est déjà crée avec le bon script

INSTALL DE BASE

```
time ansible-playbook --limit=grp-preproduction-01@grap.coop ./playbook/p_odoo_install.yml --extra-vars "{'odoo_settings': [{'name': 'odoo16_coopalim', 'branch': '16.0-COOPALIM-First-install'},]}"
```

INSTALL BD

```
sudo su odoo16_coopalim -c './env/bin/click-odoo-initdb -n coopalim_preprod_2026_04_13 --no-demo --no-cache --config ./odoo.cfg'
```

INSTALL MODULES

```
time ansible-playbook --limit=grp-preproduction-01@grap.coop ./playbook/p_odoo_update_all.yml --extra-vars "{'odoo_settings': [{'name': 'odoo16_coopalim', 'branch': '16.0-COOPALIM-First-install'},]}"
```

Pour serveur de prod

Même process, sauf que la branche est déjà crée avec le bon script

INSTALL DE BASE

```
# Sur ordi local
time ansible-playbook --limit=grp-production-08@grap.coop ./playbook/p_odoo_install.yml --extra-vars "{'odoo_settings': [{'name': 'odoo16_coopalim', 'branch': '16.0-COOPALIM-First-install'},]}"
```

INSTALL BD

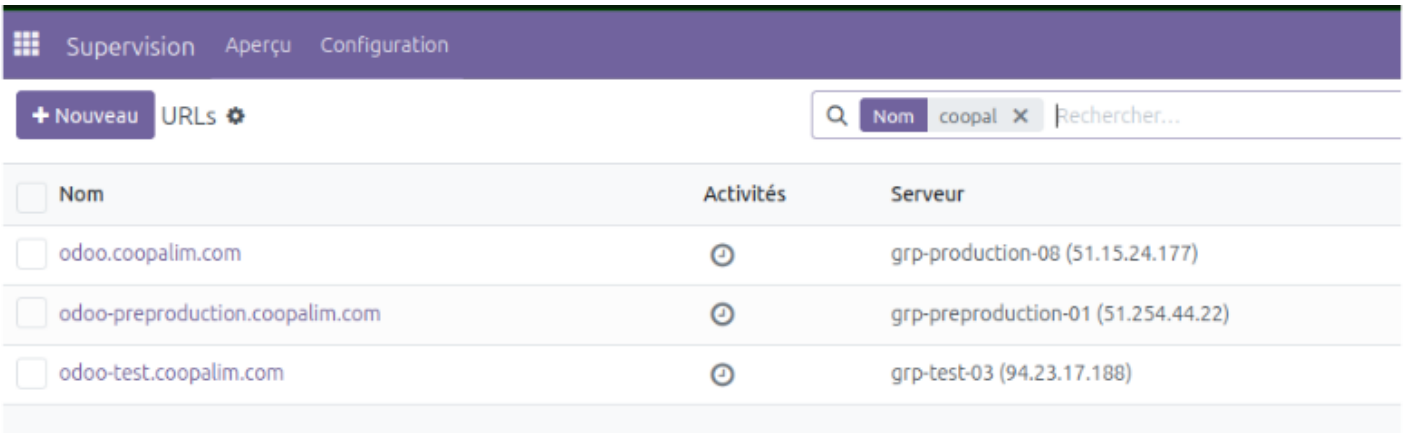
```
# Sur serveur
cd /opt/odoo16_coopalim/code
sudo su odoo16_coopalim -c './env/bin/click-odoo-initdb -n coopalim_production --no-demo --no-cache --config ./odoo.cfg'
```

INSTALL MODULES

```
# Sur Ordii local
time ansible-playbook --limit=grp-production-08@grap.coop ./playbook/p_odoo_update_all.yml --extra-vars "{'odoo_settings': [{'name': 'odoo16_coopalim', 'branch': '16.0-COOPALIM-First-install'},]}"
```

Rajouter sur https://supervision.grap.coop

Le nom de domaine et les URLS à checker

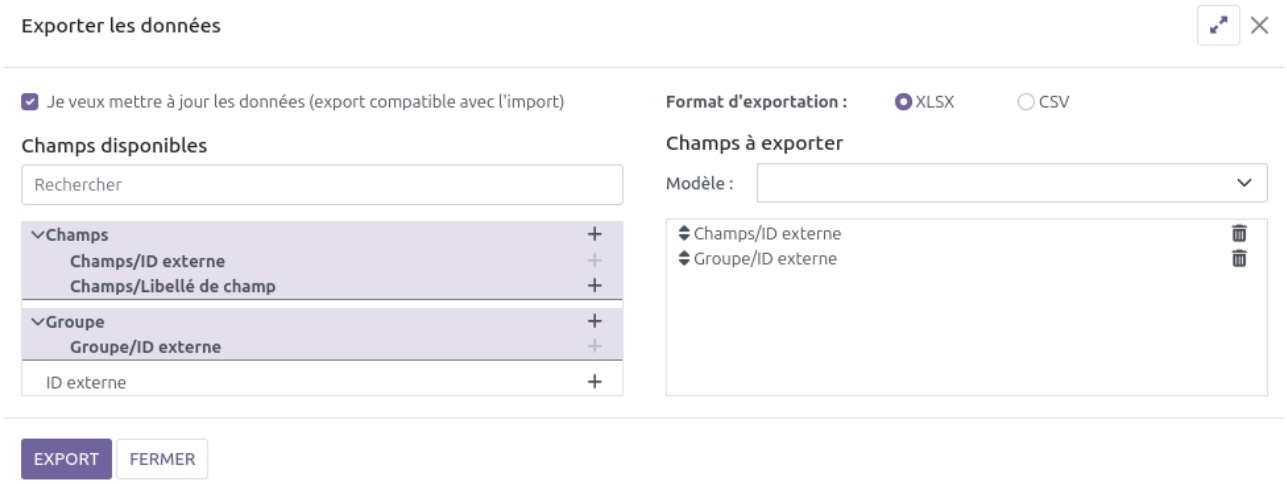


Configurations sur Odoo Prod

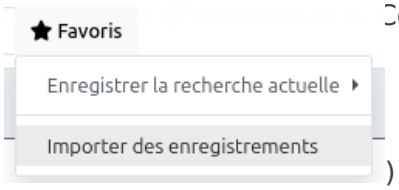
- Créer les rôles et 1ers utilisateur.ices
- Créer All et la société CAE (type Grap SCIC SA)
- Créer les (principaux) Labels / organisations certificatrices (si connues)

> NB : les conditions de paiement, le référenciel département/région/pays sont créés directement à l'initialisation de l'instance

- Dans Paramètres > Configuration réduite > lignes de configuration, préparer la configuration réduite (prendre pour modèle une autre coop), exporter ça :



et importer le fichier obtenu dans la nouvel Configuration réduite > Lignes de configuration, aller sur)



- Créer les Catégories d’articles par import => vérifier si c'est ok pour elleux de prendre le même référentiel que les autres coop : ça a l'avantage de permettre des comparaison entre coop, des stats consolidées. + voir s’ils veulent réserver certaines catégories à des groupes d'utilisateur.ices (comptables et admin 1er niveau).
- Configurer les actions serveur et éditions en lot présentes chez Grap
- Dans configuration > facturation > Configuration de transactions internes, indiquer le journal TI achat / vente, le compte TI par défaut, la position fiscale TI.

Infos à demander à la nouvelle coop

Au service compta

- Le plan comptable général France => donner à voir, et expliquer les personnalisations possibles
- Nos journaux type (cf choix des codes VT ou VTE pour vente, etc.)
- Les catégories de produits
- Les utilisations internes
- quels comptes pour le mapping de la position fiscale "transactions inter société"

Pour qu'iels donnent leurs propres codes et comptes (ex. Mache exprime ses comptes comptables avec 8 chiffres, pas Grap) + éventuelles monnaies locales.

Si le logiciel comptable n'est ni EBP, ni Sage, ni Louty, demander un exemple de fichier d'import comptable pour que l'on créé un nouveau modèle d'export depuis Odoo.

Au / à la super-utilisateur.ice

- Logo, adresse du siège, site internet, n° tel, n° TVA intracom, SIRET, APE
- Listes des personnes qui travaillent dans l'équipe support, avec photos, mails pros, tel pros, poste, ...
- Liste des organisations certificatrices

Created 2026-05-04 13:48:32 UTC by Sandie Favre
Updated 2026-08-05 09:20:37 UTC by Sandie Favre