myFAB DB Externes

Module permettant de connecter une base SQL externe, d’exécuter une requête de lecture, puis de générer automatiquement un modèle OpenProd, ses vues, son action, son menu et son mécanisme de rechargement.

** module disponible uniquement en v10 **

- Documentation utilisateurs
  - Documentation du module
  - Maintenance, bonnes pratiques et point d'attention
- Exemple de mise en oeuvre
  - Exemple
  - Utilisation dans une action serveur

Documentation utilisateurs

Documentation fonctionnelle et technique couvrant l’installation, le paramétrage des connexions, la construction SQL, la génération du modèle OpenProd, le rechargement des données et la maintenance.

Documentation du module

Présentation

1. Objectif
Le module permet d’exposer dans OpenProd des données issues d’une base SQL externe, sans développement spécifique pour chaque jeu de données.
L’utilisateur définit une connexion, saisit une requête SQL de lecture, puis laisse le module créer le modèle OpenProd et l’interface associée.

Le résultat final est un modèle manuel OpenProd alimenté à partir de la requête distante, avec sa vue liste, sa vue formulaire, son action, son menu et un cron de rechargement.

2. Principe général

- Créer une configuration de connexion SQL externe et tester l’accès.
- Créer une source de données, saisir une requête SQL et prévisualiser le résultat.
- Générer le modèle OpenProd, ses champs, ses vues et son menu.
- Charger ou recharger les données depuis la base distante vers le modèle généré.

Installation

1. Prérequis

Le module repose sur SQLAlchemy et sur les drivers Python correspondant aux connecteurs utilisés :

- psycopg2 pour PostgreSQL
- pymysql pour MySQL
- pymssql pour SQL Server

Le connecteur SQLite est également supporté.

Tous les drivers sont téléchargéq lors de l'installation du module.

2. Point d’attention

Si le driver Python correspondant au connecteur choisi n’est pas installé sur le serveur, le test de connexion et l’ouverture de session échoueront avec un message explicite.

Configuration

1. Menu de configuration

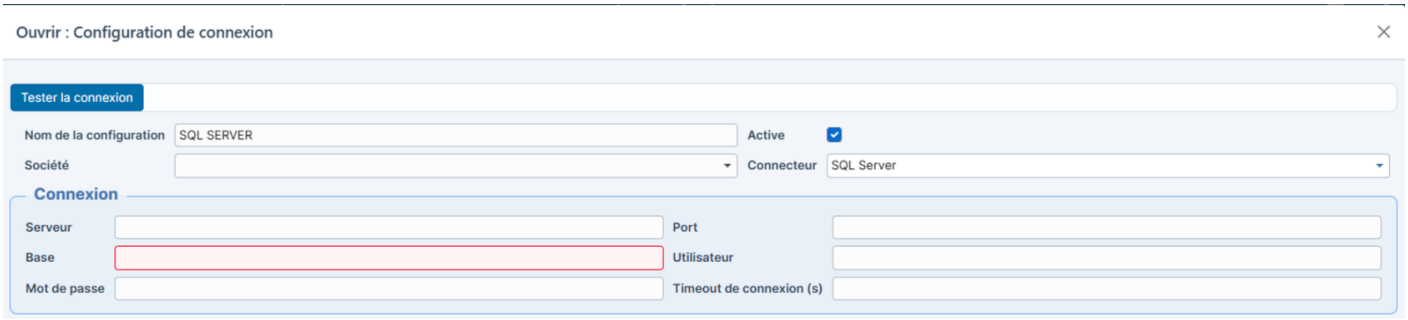
Le module ajoute un menu d’administration contenant :

- Sources de données
- Configurations de connexion

2. Configuration de connexion

Une configuration contient notamment :

- un nom unique de configuration
- un connecteur
- un serveur et un port si nécessaire
- une base cible
- un utilisateur et un mot de passe si le connecteur l’exige
- un timeout de connexion



3. Test de connexion

Le bouton Tester la connexion vérifie que la session SQL externe peut être ouverte.

Le résultat est enregistré dans un journal directement sur la configuration.

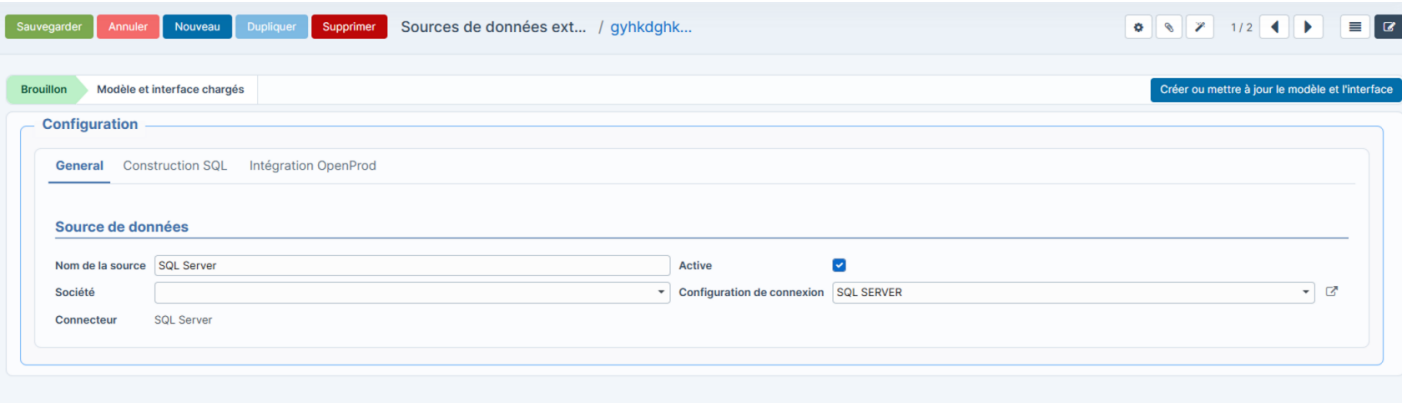
En cas d’erreur, le message remonte une information exploitable pour l’utilisateur : cible visée, timeout éventuel et détail technique retourné par le driver SQL.

Paramétrage d'une source

1. Informations principales

Une source de données associe :

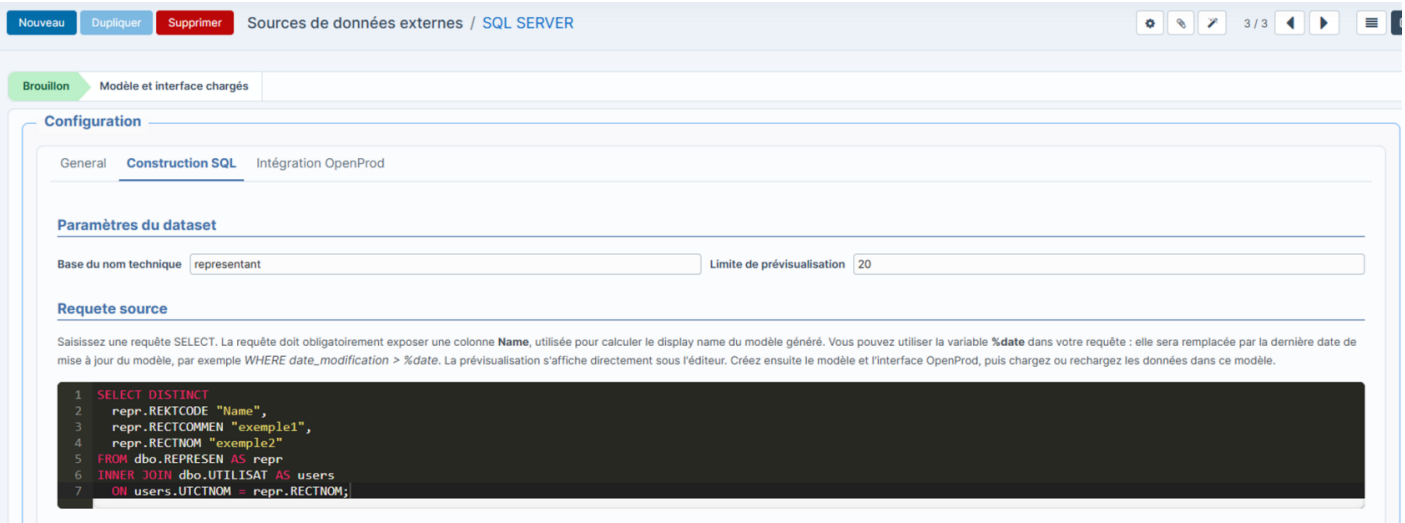
- un nom unique de source
- une configuration de connexion
- une requête SQL de lecture
- un nom technique de base pour générer le modèle OpenProd



2. Règles sur la requête SQL

La requête est validée avant exécution. Le module n’accepte que des requêtes de lecture.

- les requêtes SELECT sont autorisées
- les requêtes WITH sont acceptées si elles débouchent sur une lecture
- les mots-clés de modification comme INSERT, UPDATE ou DELETE sont refusés
- les requêtes multiples sont interdites



3. Colonne obligatoire

La requête doit obligatoirement exposer une colonne Name.

Cette colonne est transformée en champ x_name dans le modèle généré et sert de base au display_name OpenProd.

4. Variable de date

La variable %date peut être utilisée dans la requête.

Au moment de l’exécution, elle est remplacée par la dernière date de synchronisation connue.

Exemple :

SELECT Name, code, date_modification FROM ma_vue_source WHERE date_modification > %date

5. Prévisualisation

Le bouton Prévisualiser la requête exécute une version limitée de la requête et affiche le résultat directement dans la fiche.

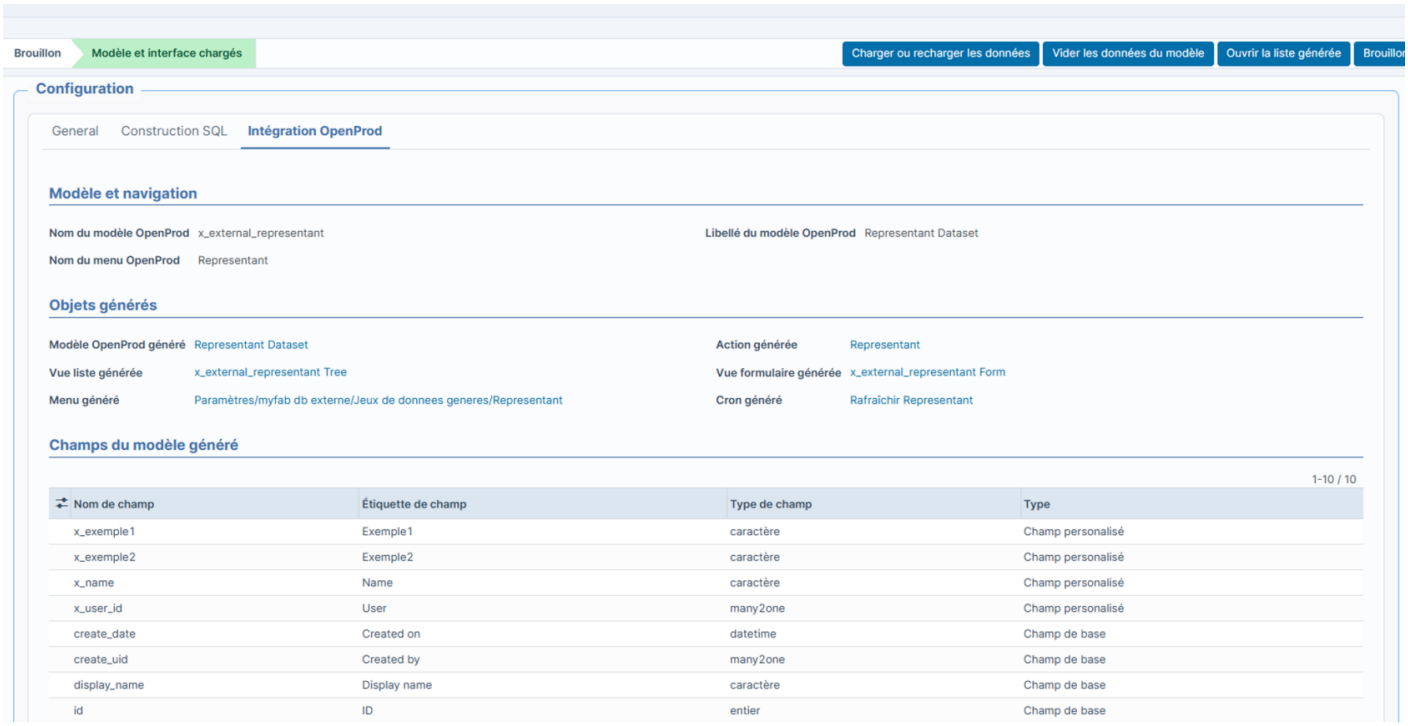
Cette étape sert à contrôler les colonnes retournées et la cohérence du jeu de données avant de générer le modèle.

6. État de la source

Le workflow de la source est volontairement simple :

- Brouillon : la requête et la structure peuvent être modifiées
- Modèle et interface chargés : le modèle existe, la requête est verrouillée, seuls les rechargements sont attendus

Pour modifier la structure après génération, il faut utiliser le bouton Brouillon, puis relancer la création ou mise à jour du modèle et de l’interface.



Utilisation du module

1. Création du modèle et de l’interface

Le bouton Créer ou mettre à jour le modèle et l’interface analyse les colonnes de la requête, crée ou met à jour le modèle manuel OpenProd, ses champs, sa vue liste, sa vue formulaire, son action, son menu et son cron de rechargement.

La conversion des colonnes suit les règles suivantes :

- Name devient x_name
- id devient x_source_id pour éviter les collisions avec l’identifiant OpenProd
- les autres colonnes deviennent des champs x_* sécurisés
- le champ technique x_user_id est ajouté pour tracer l’utilisateur créateur des enregistrements importés

2. Chargement des données

Le bouton Charger ou recharger les données exécute la requête source complète, vide les enregistrements déjà présents dans le modèle généré, puis recrée les lignes à partir du résultat courant.

Le chargement est donc un rechargement complet, et non une mise à jour ligne par ligne.

Cette stratégie simplifie la maintenance et garantit que le contenu du modèle reflète exactement le résultat SQL courant.

3. Ouverture de la liste générée

Le bouton Ouvrir la liste générée ouvre directement l’action créée pour le modèle généré, ce qui permet de consulter les données importées comme n’importe quel objet OpenProd.

4. Vidage des données

Le bouton Vider les données du modèle supprime les lignes importées sans supprimer le modèle, ses champs ni son interface.

5. Rafraichir les données automatiquement

Pour rafraichir les données deux options s'offrent à vous :

- Utiliser le bouton d'action de rafraichissement des données présent dans la vue formulaire
- Activer le cron créé automatiquement en même temps que le modèle et lui donner une valeur temporelle.

Maintenance, bonnes pratiques et point d'attention

Maintenance et bonne pratique

1. Évolution du schéma SQL
Si la requête évolue et change de colonnes ou de types, il faut repasser la source en Brouillon, puis relancer la génération du modèle et de l’interface avant tout nouveau rechargement.

2. Verrouillages de sécurité

- le nom des configurations de connexion doit être unique
- le nom des sources doit être unique
- la requête SQL ne peut plus être modifiée lorsque l’état n’est plus Brouillon
- les champs structurants du modèle généré sont verrouillés une fois l’interface créée

3. Suppression d’une source
La suppression d’une source supprime également les objets OpenProd générés qui lui sont associés : modèle manuel, données, vues, action, menu, cron et droits d’accès générés par le module.

4. Recommandations

- privilégier des requêtes SQL lisibles, stables et limitées au besoin métier réel
- conserver un alias Name explicite dans la requête pour obtenir des libellés utilisateurs clairs
- tester systématiquement la connexion et la prévisualisation avant la première génération
- utiliser %date uniquement si une première synchronisation a déjà été réalisée

Point d'attention

Le module utilise des drivers python afin de se connecter aux différents système de gestion de base de données. Veuillez donc vous assurer d'avoir une version de votre système géré pas un des drivers suivant.

SQLAlchemy (Python)

1. Python 3.x: 3.7 ou plus récent

PyMySQL (MySQL)

1. Python 3.x: 3.7 ou plus récent
2. Mysql 5.7 ou plus récent
3. MariaDB 10.4 ou plus récent

Pymssql (SQL Server)

1. Python 3.x: 3.7 ou plus récent
2. FreeTDS 1.4.10 ou plus récent
3. Cython 3.0.7 ou plus récent
4. Microsoft SQL Server 2005 ou plus récent

Charge système

> Le rafraîchissement des tables créées étant géré par des tâches planifiées (crons), l'activation d'un grand nombre de crons peut augmenter la charge système et ralentir le serveur. Il est donc recommandé d'éviter l'exécution simultanée de nombreux crons ou de configurer des fréquences de rafraîchissement trop élevées.

Modification de schéma

> Dans le cas où vous repassez un enregistrement au brouillon afin de modifier son schéma (suppression ou ajout de colonnes), les vues personnalisées créées ainsi que les actions serveur utilisant le modèle généré peuvent devenir obsolètes. Cela peut notamment être dû à un changement de nom d'un champ ou à sa suppression.

Exemple de mise en oeuvre

Exemple de mise en place d’une source de données externe alimentant un modèle OpenProd à partir d’une table SQL distante contenant des clients ou des tiers.

Exemple

1. **Créer la configuration de connexion**
   Créer une configuration nommée par exemple ERP secondaire - PostgreSQL ou SQL Server avec :

- Connecteur : PostgreSQL
- Serveur : 10.0.0.15
- Port : 5432
- Base : erp_reporting
- Utilisateur : compte de lecture dédié

- Connecteur : SQL Server
- Serveur : 10.0.0.16
- Port : 1433
- Base : erp_exemple
- Utilisateur : compte de lecture dédié

2. Créer la source de données
Créer une source nommée Clients externes puis associer la configuration précédente.

3. Saisir la requête SQL
Exemple :
SELECT customer_name AS Name, customer_code AS test, city AS test2, last_update FROM customer_export WHERE active = true

Dans cet exemple, la colonne Name alimente automatiquement le champ x_name du modèle généré.
Les colonnes test et test2 deviendront respectivement x_test et x_test2.

4. Prévisualiser
Utiliser le bouton Prévisualiser la requête pour vérifier que les colonnes exposées, les libellés et les lignes retournées correspondent au besoin.

5. Générer le modèle OpenProd
Cliquer sur Créer ou mettre à jour le modèle et l'interface.
Le module crée alors un modèle manuel du type x_external_clients_externes avec les champs nécessaires et l'interface OpenProd associée.

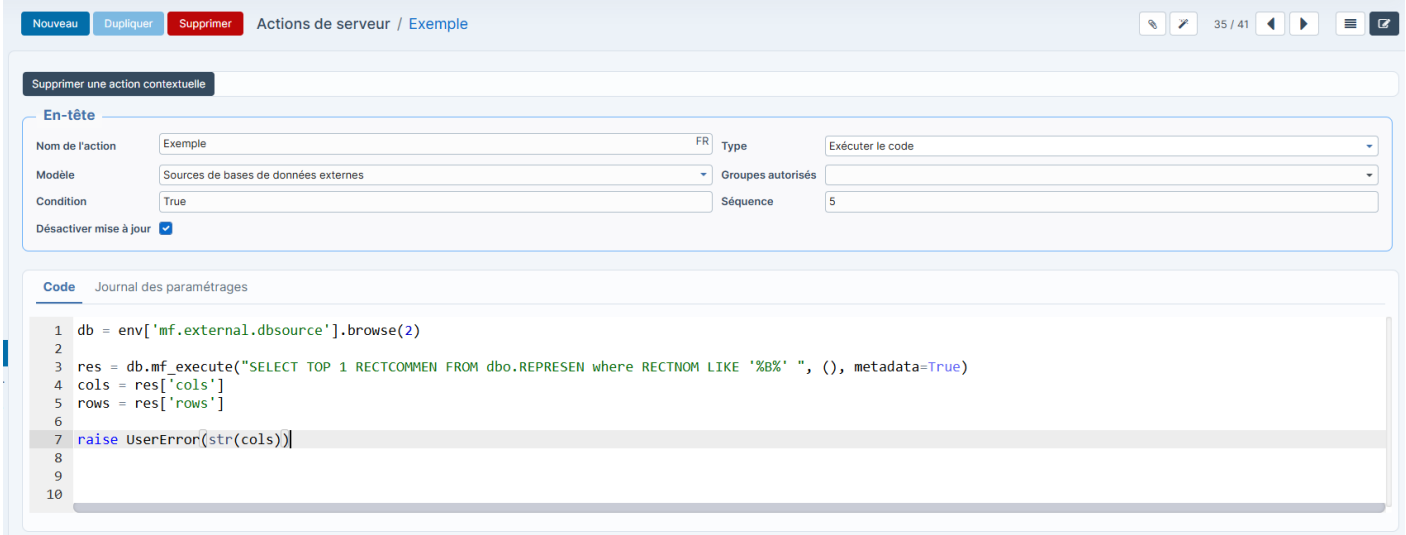
6. Charger les données
Cliquer sur Charger ou recharger les données pour importer les lignes dans le modèle généré, puis ouvrir la liste avec le bouton dédié.

7. Variante incrémentale
Si la table source contient une date de modification, la requête peut être adaptée avec %date après une première synchronisation réussie.

Utilisation dans une action serveur

Il est possible d'utiliser un enregistrement du module directement dans une action serveur.

La méthode à utiliser est mf_execute, qui attend comme paramètres une requête SQL via query, des paramètres d'exécution optionnels via execute_params, et un booléen metadata permettant de préciser si le retour doit inclure uniquement les lignes ou également les métadonnées de colonnes.



Côté Openprod, pour faire de la lecture en test il est recommandé d'utiliser la méthode UserError qui permet d'afficher soit les noms des colonnes, soit leur contenu. Ici avec les variables cols et rows.

Affichage de la variable 'cols' :

