



# Séminaire QGIS utilisateur 2107 Montpellier

---



# Qui sommes-nous?

## Oslandia

- Open Source pure player
- Solutions géospatiales
- 3 développeurs QGIS « core committers



14 employés en télétravail partout en France





# QGIS 3 ... ARRIVE

# ÉNORMÉMENT de nouvelles fonctionnalités !

Processing becoming an ETL

Processing refactored in C++

No more shp  
limitations in  
Processing

GPKG is now the  
default format !

**3D view**

Map Composer fully rewritten  
→ LAYOUTS !

New productive editing tools  
node tool, 3D handling, CAD inspired tools,  
advanced dock for existing features, new snapping dock

Unified data access button

User profile  
switcher

Metadata editor  
geonode integration

**Native multi Canvas**

Full QGIS server Refactor  
WMS 100 % OGC Compliant

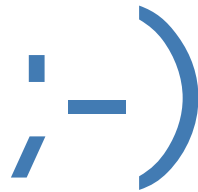
Raster data editable

Unified search bar

# C'est beaucoup..

---

Parlons plutôt de choses un peu cachées



# Coder en base « épaisse » alias thick database

## Garder la logique métier dans la base de données ?

Vues, triggers, procédures stockées comme interface / API / services

Les groupes de transaction QGIS sont fait pour cela !

Chaque modification est envoyée à la base directement

Toutes les couches du même groupe de transaction sont évaluées ensemble

- 9 Réactivité utilisateur, contraintes évaluées à la volée



 Undo Redo




**Réseaux opérateurs**

   **Consignations**

## Interventions




**Evenements**

Ouvrages

☒ ☐ Bâche


☒ BE (Branchement d'égout)

☒ BP (Branchement particulier)

☒  Rejets BP [0]

☒ ☐ : BR (Branchement regard)

☒ ☐ **Décanteur**☒ Échangeur [0]

☒  Glissière [0]

☒  Groupe actionneur

☒ ☐ Niches

The map shows the study area with the following features:

- Streets:** Rue De Palestro, Passage Basfour, and Rue De Palestro.
- Sampling Points:** 9 (blue circle) and 22 (blue square).
- Building:** RC (...66267) (blue square).
- Scale:** 0 to 6 meters.
- Blue Line:** Connects the sampling points 9 and 22.

Mais la pile ANNULER/REFAIRE est vide

---

→ QGIS Grant application (financement QGIS.org)

UNDO REDO de retour, les modifications sont stockées  
DANS PostgreSQL sous forme de SAVEPOINTS.

>>> Les transactions groups ne sont plus expérimentaux





# Utiliser les procédures stockées - groupes de transaction

Exemple :

« SELECT \* from inverser\_troncon(troncon\_id) »

OU

« SELECT \* from decouper\_troncon(troncon\_id, clic\_x, clic\_y) »

```
1  -- Function: api_maj.decouper_troncon(geometry, uuid, real)
2
3  -- DROP FUNCTION api_maj.decouper_troncon(geometry, uuid, real);
4
5  CREATE OR REPLACE FUNCTION api_maj.decouper_troncon(
6      split_geom geometry,
7      tr_id uuid DEFAULT NULL::uuid,
8      tolerance REAL DEFAULT 0.1)
9      RETURNS uuid AS
10     $BODY$
11     DECLARE
12         var_t_id uuid;
13         tabs REAL;
14     BEGIN
15
16     IF tr_id IS NULL THEN
17         --select into t_id, tabs troncon_id, abscisse from api_maj.accrocher_a_troncon(split_geom);
18         SELECT t_id, abscisse INTO var_t_id, tabs FROM api_maj.accrocher_a_troncon(split_geom);
19     IF var_t_id IS NULL THEN
20         RAISE EXCEPTION 'Tronçon d''attache introuvable';
```

# ...dans des actions QGIS (par exemple)

Propriétés de la couche - Tronçons | Actions

▼ Liste d'actions

Type	Description	Titre Court	Action	Capture	Afficher o
Python	Tronçon - ouvrir le formul...		registry =...	<input type="checkbox"/>	<input type="checkbox"/>
Python	Tronçon - Inverser sens		# coding:...	<input type="checkbox"/>	<input type="checkbox"/>
Python	Tronçon - Découper		# coding:...	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Python					
Python					
Python					

**Éditer l'action**

Type: Python ☐ Rendu de capture ☐ Afficher dans la table d'attributs

Description: Tronçon - Inverser sens

Nom court: Laisser vide pour utiliser uniquement une icône

Icône:

**Texte de l'action**

Le texte de l'action définit ce qui se produit lorsque l'action est lancée.  
Le contenu dépend du type d'action.  
Pour le type *Python* le contenu doit être du code Python.  
Pour les autres types, il doit être un fichier ou une application avec des paramètres optionnels.

```
16 request = QgsFeatureRequest().setFilterFid(fid)
17 feature = next(layer.getFeatures(request))
18 fake_value = feature.attribute(fake_field_name)
19 fake_field_id = feature.fieldNameIndex(fake_field_name)
20 layer.editBuffer().changeAttributeValue(fid, fake_field_id, fake_
21
22 # trigger the stored procedure doing the real work
23 sql = 'SELECT api_maj.inverser_sens_ecoulement(\'\' + tr_id + '\')';
24 transaction = layer.dataProvider().transaction()
25 transaction.executeSql(sql)
26
27 # force redraws
```

Identifier

Tronçon - ouvrir le formulaire

Tronçon - Inverser sens

Tronçon - Découper

Tronçon - Déplacer

Annuler OK

# MAIS parfois QGIS ne pouvait pas sauver ces modifications

Rien n'indique à QGIS que

« `QgsTransaction.ExecuteSQL('monSQL')` »

a modifié la couche (ou une couche)

→ ajout d'un flag et d'un nom de transaction

« `QgsTransaction.ExecuteSQL('monSQL', LayerModified, actionName)` »



# Un peu plus de FUN?

---

QGIS 3 permet de rafraîchir à intervalle régulier une couche

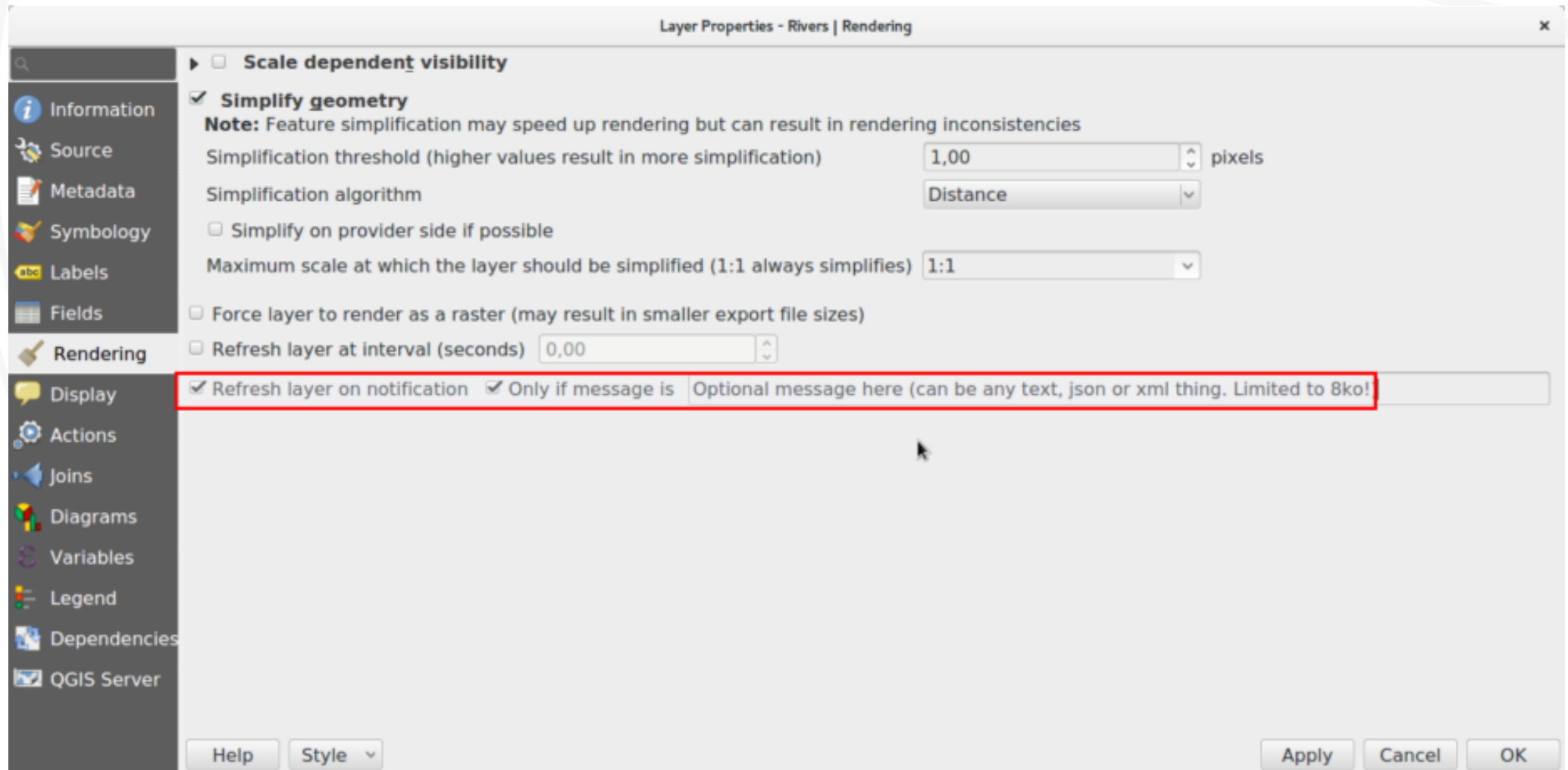
Pratique (données vivantes, temps réel, tableau de bord, etc..) !

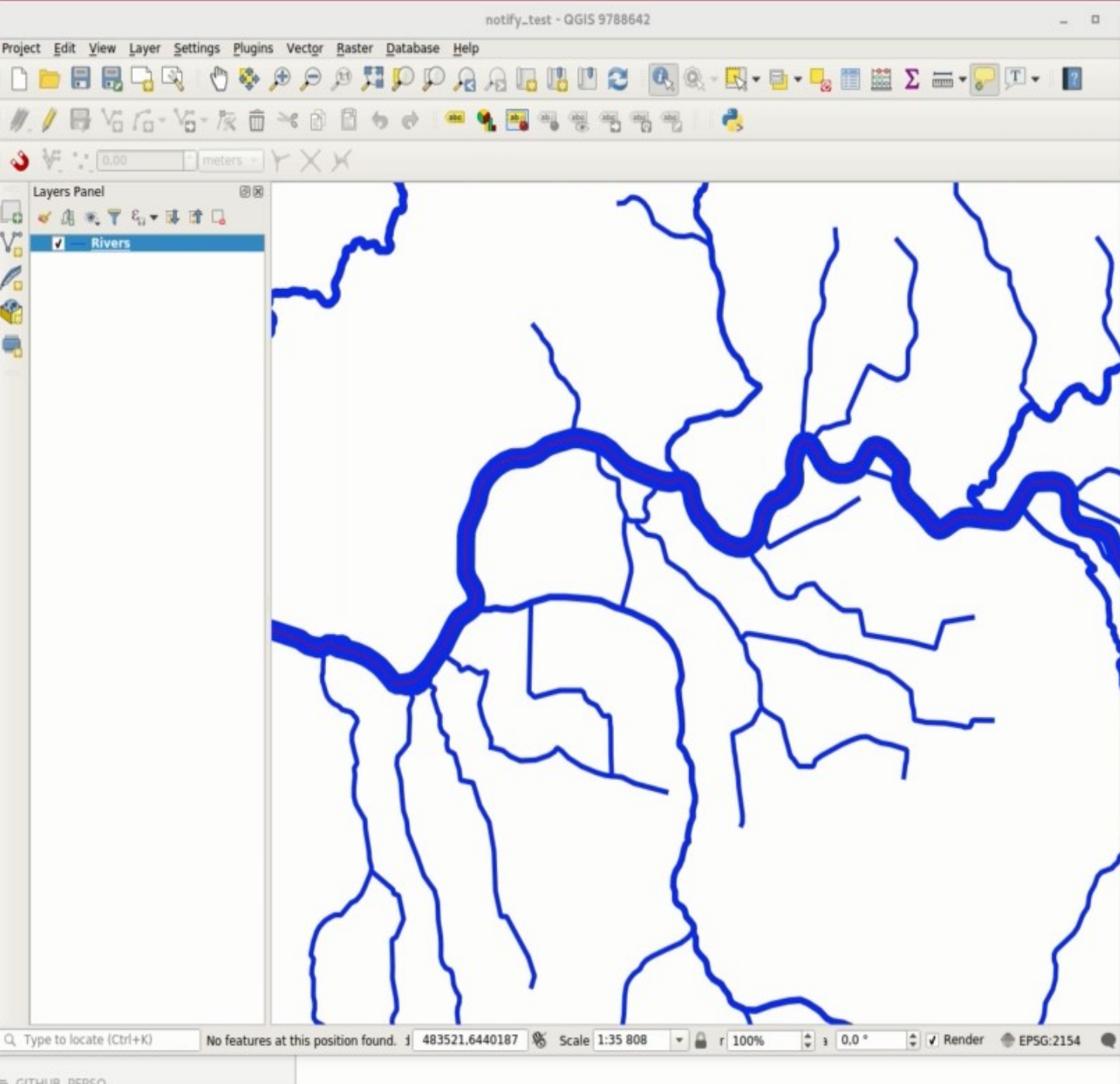
Mais consommateur, surtout pour des requêtes volumineuses

Exploiter les NOTIFY CHANNELS !



# QGIS 3 – PG Notify





```
regis@lenovo-X1: ~/APPS/QGIS3
regis@lenovo-X1:~/APPS/QGIS3$ psql -d refgeo2
psql (9.6.5, serveur 9.5.9)
Saisissez « help » pour l'aide.

refgeo2=# NOTIFY qgis, 'refresh qgis';
NOTIFY
refgeo2=# UPDATE ref.hydro_cours_eau set
classification = 2, geoml93 = st_simplify(geoml93,500) where code_cours_eau = 'P56157';
```





# Potentiellement peut déclencher toute action

Add New Action

TypeGenericCapture outputDescriptionMandatory descriptionShort NameLeave empty to use only iconIconAction ScopesField ScopeFeature ScopeCanvasLayer ScopeAction textThe action text defines what happens if the action is triggered. The content depends on the type. For the type Python the content should be python code For other types it should be a file or application with optional parameters1InsertExecute if notification matches

HelpCancelOK

Ou comment stocker des données dans le projet QGIS

- étiquetage personnalisé
- symbologie sur mesure
- source de données non modifiable / ne pas mélanger représentation et données



## Memory Layer :

non persistant

## Extension Memory Layer Saver :

fichier .mldata à côté du projet

## Etiquetage personnalisé :

modification obligatoire de la source de données

gros travail de calibration

Plugin EasyCustomLabeling & AI.

Duplication des données

Perte du lien avec la source

Perte de l'intelligence du moteur d'étiquetage (conversion en centroïdes)

## Ajout d'un format de stockage de données associé au projet

SQLITE (provider OGR). Léger et rapide

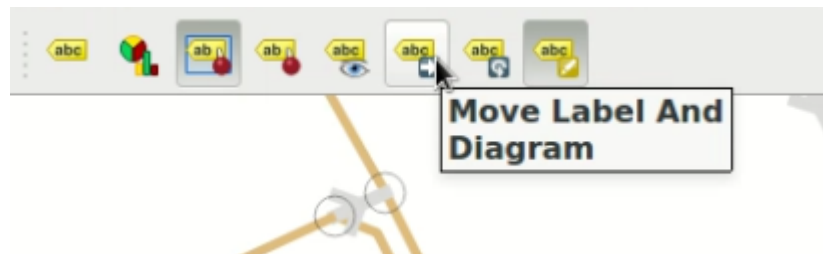
## Ajout d'un format de projet .QGZ (zip) optionnel

## Ajout des jointures éditables

INSERT OR UPDATE = UPSERT

# QGIS 3 – stockage auxiliaire

Activation par défaut de la barre d'étiquetage

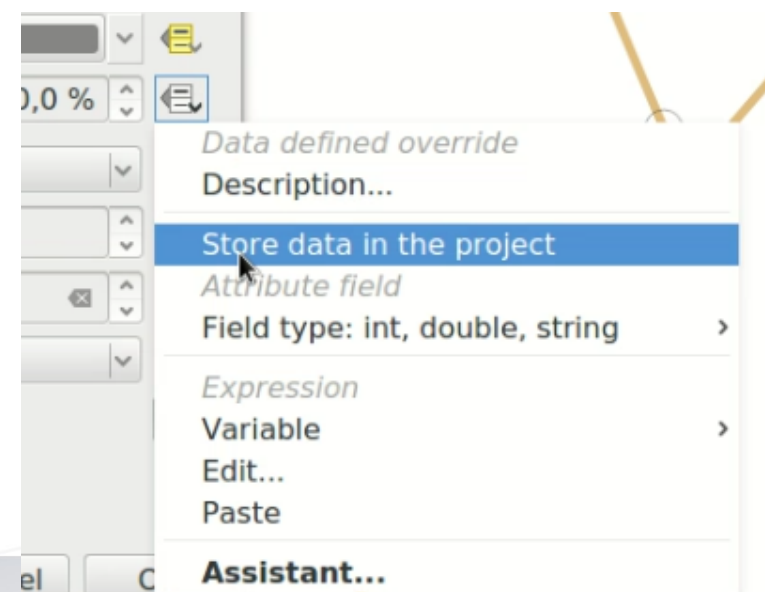


Création automatique d'une base .qld de stockage

Ajout automatique de champs personnalisés liés à l'étiquetage

Ajout simplifié de tout autre champ

Maintenance automatique avec la couche d'origine  
(pour les éditions faites depuis QGIS)



# Nouvel onglet d'administration des données auxiliaires

Layer Properties - Tronçons | Auxiliary Storage

Auxiliary storage tables can contain additional data that should only belong to the project file. For instance, specific location or rotation for labels. Auxiliary data are saved in qgd files. New fields can be added from any data-defined widget when needed. Be aware that this information will NOT be saved in your datasource but only in the project file.

▼ **Information**

Key

Features

Fields

▼ **Fields**

Target	Property	Name	Type	Full Name
labeling	Hali		String	labeling_hali
labeling	Vali		String	labeling_vali
labeling	ScaleVisi...		Integer	labeling_scalevisibility
labeling	MinScale		Real	labeling_minscale
labeling	MaxScale		Real	labeling_maxscale
labeling	AlwaysSh...		Integer	labeling_alwaysshow

+

-

Auxiliary Layer ▼

- Create
- Clear
- Delete
- Export

16 000 lignes de code !

De nombreux effets de bords positifs !

Jointures éditables

Stockage de ressources dans le QGZ (symboles, polices, données..)

Il manque encore :

Les connecteurs d'étiquette (flèches) natives

Les chemins d'étiquettes (courbe, multi-ligne, etc.)

Les étiquettes alternatives (substitution en cas de manque de place)

Toute contribution sera la bienvenue !