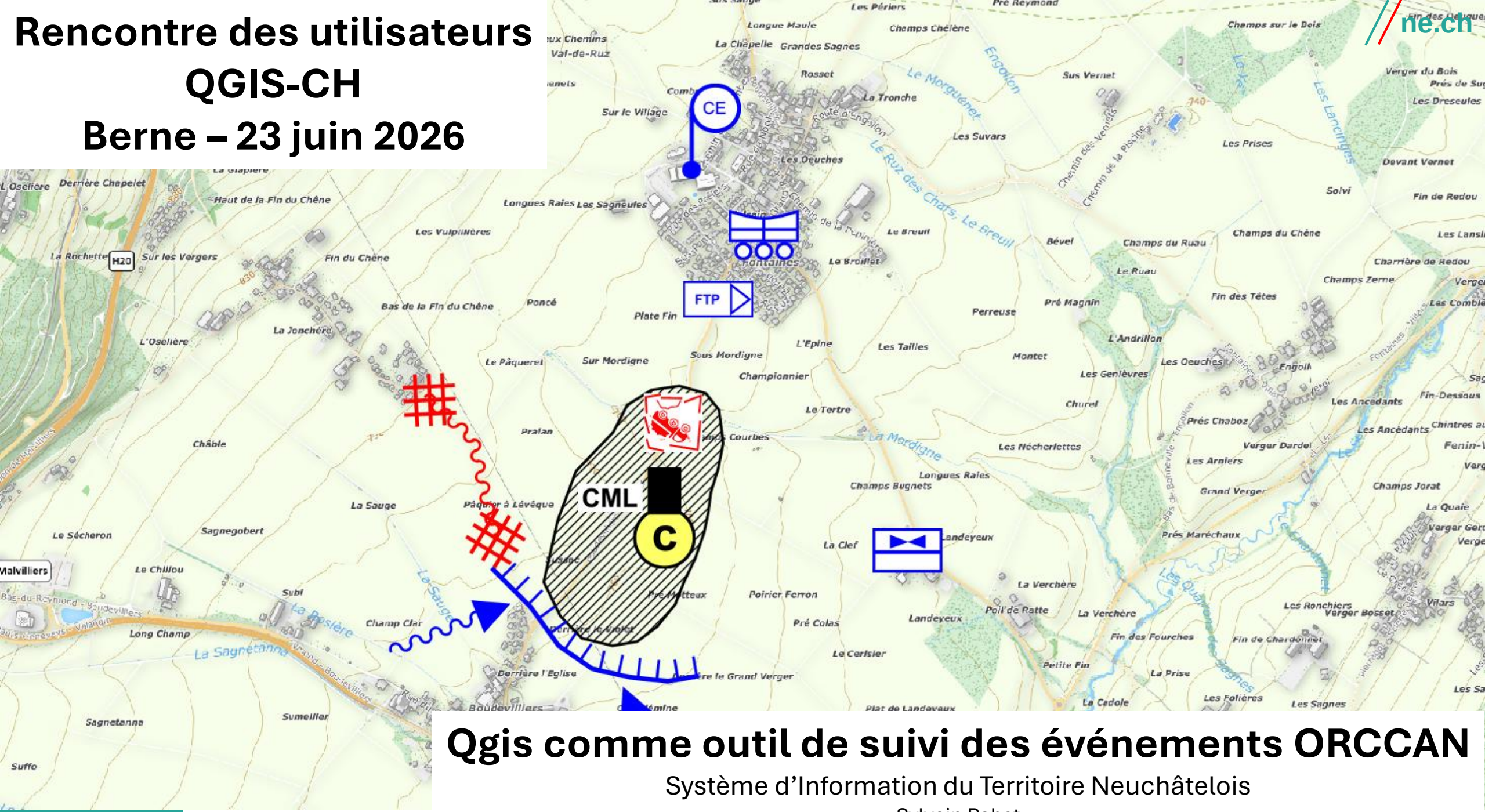
# Rencontre des utilisateurs QGIS-CH
# Berne – 23 juin 2026


CE
FTP
CML
C
Fontaines
Plate Fin
Malvilliers
Boudevilliers
Sagnetanna
Suffo
Sumeiller
Le Sécheron
Sagnegobert
Le Chillou
Subi
Long Champ
Champ Clar
La Sagnétanne
La Sauge
Pâquier à Lévêque
Pralan
Châble
Derrière l'Eglise
Derrière le Grand Verger
Pré Matteux
Poirier Ferron
Pré Colas
Le Cerisier
Landeyeux
La Clef
Champ Bugnets
Longues Raies
Les Nécherlettes
Les Arniers
Grand Verger
Prés Maréchaux
La Verchère
Poil de Ratte
Fin des Fourches
Petite Fin
La Prise
La Cedole
Les Folières
Les Sagnes
Les Ronchiers
Verger Bosset
Vilars
Champ Jorat
La Quaie
Les Ancédants
Prés Chaboz
Verger Dardel
Les Genièvres
Les Oeuches
Montet
Perreuse
Pré Magnin
Fin des Têtes
Champs du Chêne
Champs du Ruau
Bével
Le Ruau
Champs Zerne
Charrière de Redou
Fin de Redou
Solvi
Devant Vernet
Les Prises
Les Suvars
Sus Vernet
Verger du Bois
Les Dresceules
Champs sur le Bois
Champs Chélène
Grandes Sagnes
La Chapelle
Rosset
La Tronche
Sur le Village
Les Oeuches
Le Breuil
Le Broillet
Les Sagneules
Longues Raies
Les Vulpillères
Fin du Chêne
Haut de la Fin du Chêne
Derrière Chapelet
Sur les Vergers
Bas de la Fin du Chêne
Poncé
La Jonchère
L'Oselière
Le Pâquerel
Sur Mordigne
Sous Mordigne
L'Epine
Les Tailles
Championnier
Le Tertre
Les Courbes
Pré Colas
Champ Clar
Le Chillou
Sagnegobert
Le Sécheron
Malvilliers
Sagnetanna
Suffo
Sumeiller
Boudevilliers
Derrière l'Eglise
Derrière le Grand Verger
Pâquier à Lévêque
La Sauge
Pralan
Châble
Poirier Ferron
Pré Matteux

# Déroulé de la présentation

- Attentes de l’application
- Éléments de base
- Principes de fonctionnement
- Filtre des événements
- Point – Ligne – Zone
- Publication de l'événement sur le SITN
- Archiver l'événement
- Couches SITN – Fonds de plan
- Démo
- Suite

## L’application n’est pas :

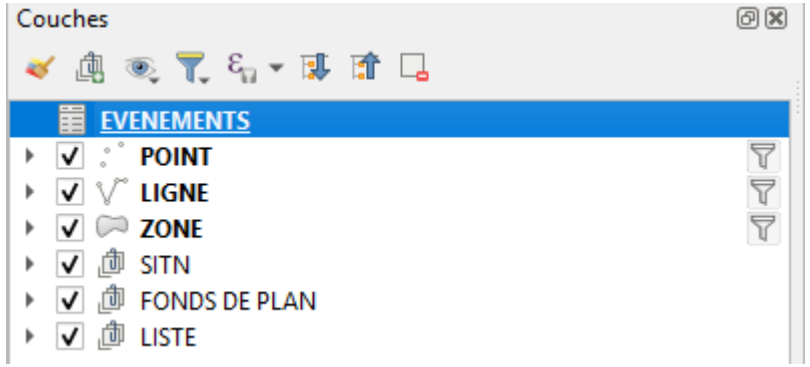
- un journal de commandement

## L’application doit :

- pouvoir gérer plusieurs événements simultanés
- conserver l’historique (temporalité des objets)
- accéder aux diverses données du SITN ou autre
- pouvoir projeter des cartes dans un poste de commandement
- pouvoir imprimer des cartes
- pouvoir transmettre les informations aux forces engagées (Police, Pompier, Sanitaire, CNU)
- disposer d’une version hors ligne
- être utilisable par des personnes sans connaissances particulières en géomatique et outils SIG

# Éléments de base

- Base de données PostGIS (hors ligne Geopackage)
- Trigger (PG) pour assurer la cohérence des données (dates, suppressions)
- 3 types de User (administrateurs, utilisateurs et lecteurs)
- Tous les styles ainsi que le projet sont stockés dans la base de données PG ou GPKG
- 4 couches modifiables
  - 1 couche Événement (non géométrique)
  - 3 couches (selon type de géométrie)
- Données du SITN (PG ou GPKG)
- Fonds de plan WMS (à voir pour hors ligne)
- Listes diverses
- Actions Qgis



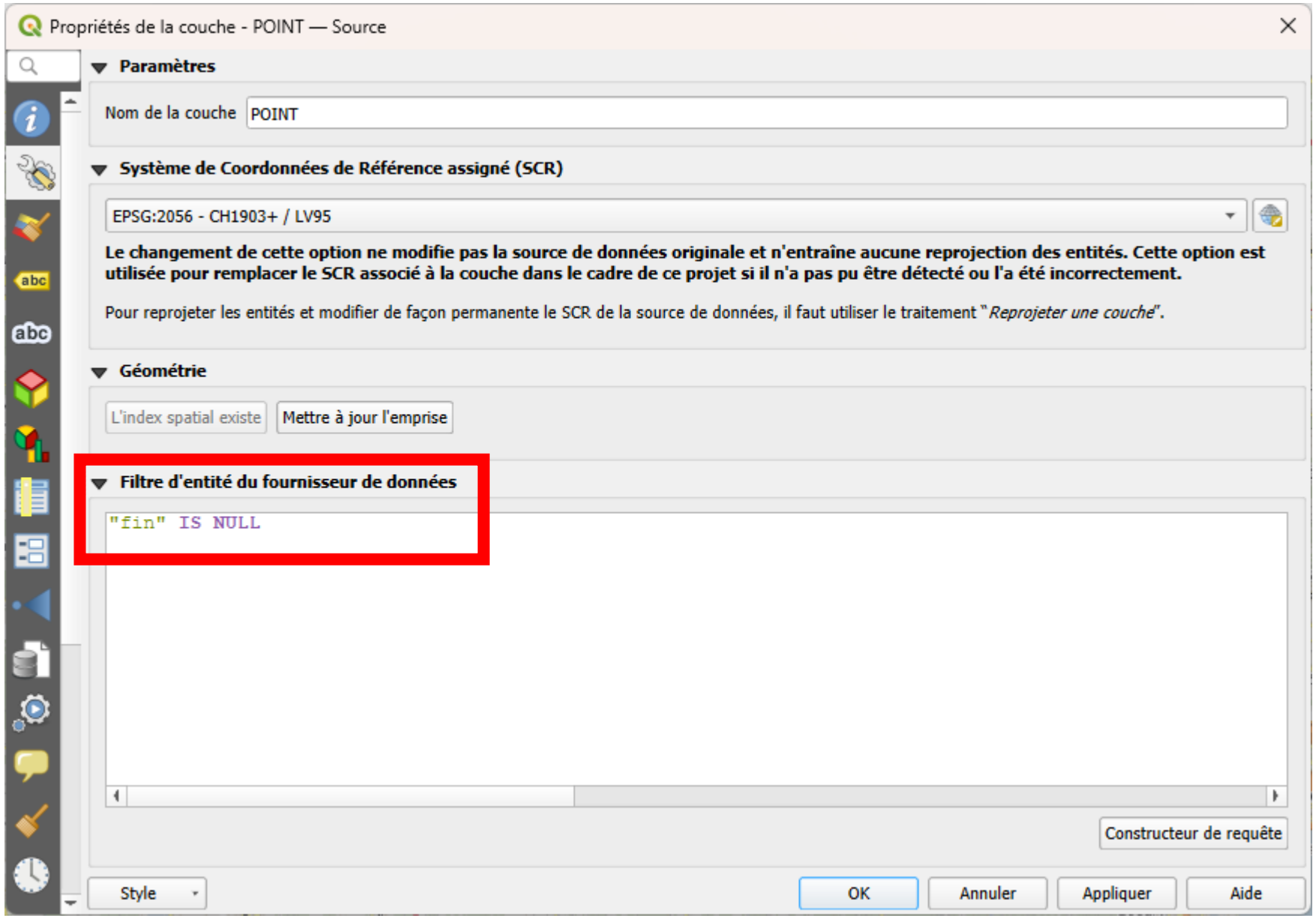


# Principes de fonctionnement

- Création d’un événement
- Application d’un filtre sur cet événement
- Insertion, mise à jour et suppression des objets Point – Ligne – Zone
- Seuls les objets en cours sont affichés
- Publication sur le guichet cartographique
- Fin de l’événement et archivage

# Filtre

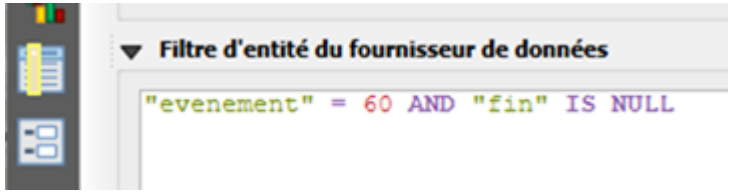
- De base le filtre sélectionne uniquement les objets en cours



# Filtre

- L'action «Activer» complète le filtre avec l'identifiant de l'événement

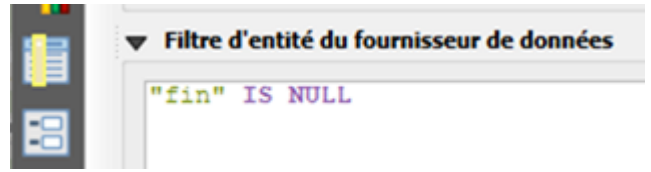
```
from qgis.PyQt.QtWidgets import QMessageBox
from qgis.core import QgsProject, QgsVectorLayer

# Couche et état du formulaire courant
couches = QgsProject.instance().mapLayersByName("EVENEMENTS")
if couches:
    couche_evt = couches[0]
    evt_etait_en_edition = couche_evt.isEditable()
else:
    QMessageBox.warning(None, "Erreur", "Couche EVENEMENT introuvable")

# Données du formulaire
current_feature = form.currentFormFeature()
id_evt_raw = current_feature.attribute('idobj')
nom_evt = current_feature.attribute('nom')

# Conversion sécurisée de l'ID
try:
    id_evt = int(id_evt_raw)
except (TypeError, ValueError):
    QMessageBox.warning(None, "Erreur", f"Identifiant invalide : {id_evt_raw}")
    raise

# Mise à jour de la variable de projet
variables = QgsProject.instance().customVariables()
variables["evenement_id"] = str(id_evt)
QgsProject.instance().setCustomVariables(variables)

etats_edition = {}

# Liste des noms des couches à filtrer
noms_couches_a_filtrer = ["POINT", "LIGNE", "ZONE"]

# Sauvegarder l'état et fermer l'édition
for nom_couche in noms_couches_a_filtrer:
    couches = QgsProject.instance().mapLayersByName(nom_couche)
    if not couches:
        continue # couche non trouvée
    couche = couches[0]  # on prend la première trouvée
    etats_edition[couche.id()] = couche.isEditable()

    if couche.isEditable():
        couche.commitChanges()

# Appliquer le filtre uniquement sur les couches point, ligne, zone
    if isinstance(couche, QgsVectorLayer):
        if "evenement" in [f.name() for f in couche.fields()]:
            # Appliquer un filtre sut l'id de l'événement et fin IS NULL
            couche.setSubsetString(f'"evenement" = {id_evt} AND "fin" IS NULL')

# Réactiver l'édition uniquement sur les couches qui l'étaient
for nom_couche in noms_couches_a_filtrer:
    couches = QgsProject.instance().mapLayersByName(nom_couche)
    if not couches:
        continue

    couche = couches[0]
    if etats_edition.get(couche.id(), False):
        couche.startEditing()

# Réactiver l'édition de la couche EVENEMENTS si elle l'étaient
if evt_etait_en_edition and not couche_evt.isEditable():
    couche_evt.startEditing()

# Rafaîchir
couche_evt.triggerRepaint()
form.setFeature(form.currentFormFeature())

# Message de confirmation
QMessageBox.information(None, "INFO", f"Le filtre est activé pour l'événement : {nom_evt}")
```



# Filtre

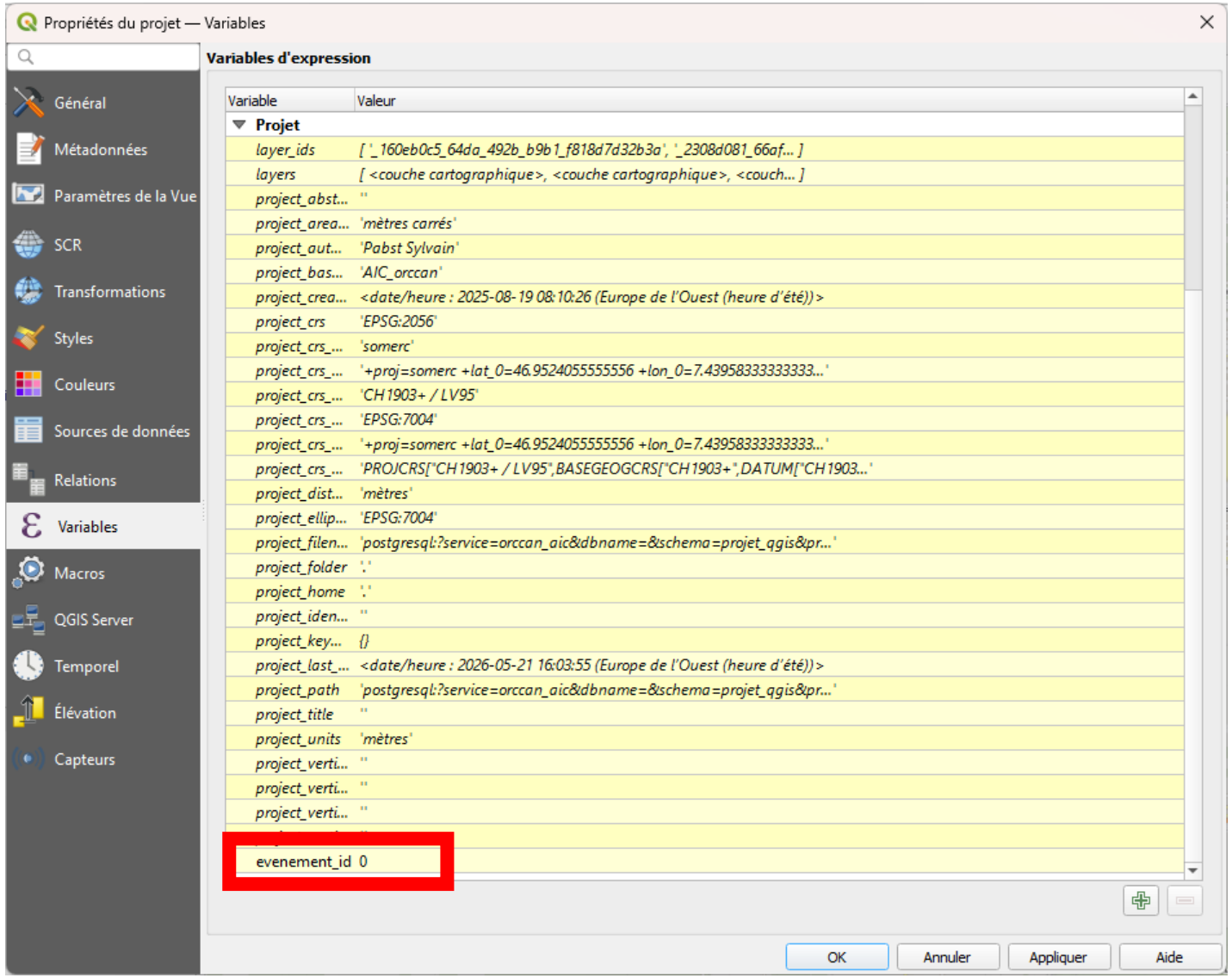
- L’action «Désactiver» supprime l’identifiant de l’événement

```
from qgis.PyQt.QtWidgets import QInputDialog
from qgis.PyQt.QtWidgets import QMessageBox
from qgis.core import QgsProject
from qgis.utils import iface

# Couche et état du formulaire courant
couches = QgsProject.instance().mapLayersByName("EVENEMENTS")
if couches:
    couche_evt = couches[0]
    evt_etait_en_edition = couche_evt.isEditable()
else:
    QMessageBox.warning(None, "Erreur", "Couche EVENEMENT introuvable")

# Supprimer le filtre = vider la variable
variables = QgsProject.instance().customVariables()
variables["evenement_id"] = 0
QgsProject.instance().setCustomVariables(variables)

etats_edition = {}

# Liste des noms des couches à filtrer
noms_couches_a_filtrer = ["POINT", "LIGNE", "ZONE"]

# Sauvegarder l'état et fermer l'édition
for nom_couche in noms_couches_a_filtrer:
    couches = QgsProject.instance().mapLayersByName(nom_couche)
    if not couches:
        continue

    couche = couches[0]
    etats_edition[couche.id()] = couche.isEditable()
    if couche.isEditable():
        couche.commitChanges()

# Appliquer le filtre uniquement sur ces couches
    if couches:
        couche = couches[0]
        couche.setSubsetString('"fin" IS NULL')

#  Réactiver l'édition uniquement sur les couches qui l'étaient
for nom_couche in noms_couches_a_filtrer:
    couches = QgsProject.instance().mapLayersByName(nom_couche)
    if not couches:
        continue

    couche = couches[0]

    if etats_edition.get(couche.id(), False):
        couche.startEditing()

# Réactiver l'édition uniquement sur si elle l'étaient
if evt_etait_en_edition and not couche_evt.isEditable():
    couche_evt.startEditing()

# Rafraîchir le formulaire
couche_evt.triggerRepaint()
form.setFeature(form.currentFormFeature())

# Message de confirmation
QMessageBox.information(None, "INFO", f"Il n'y a plus de filtre actif, tous les objets sont affichés")
```

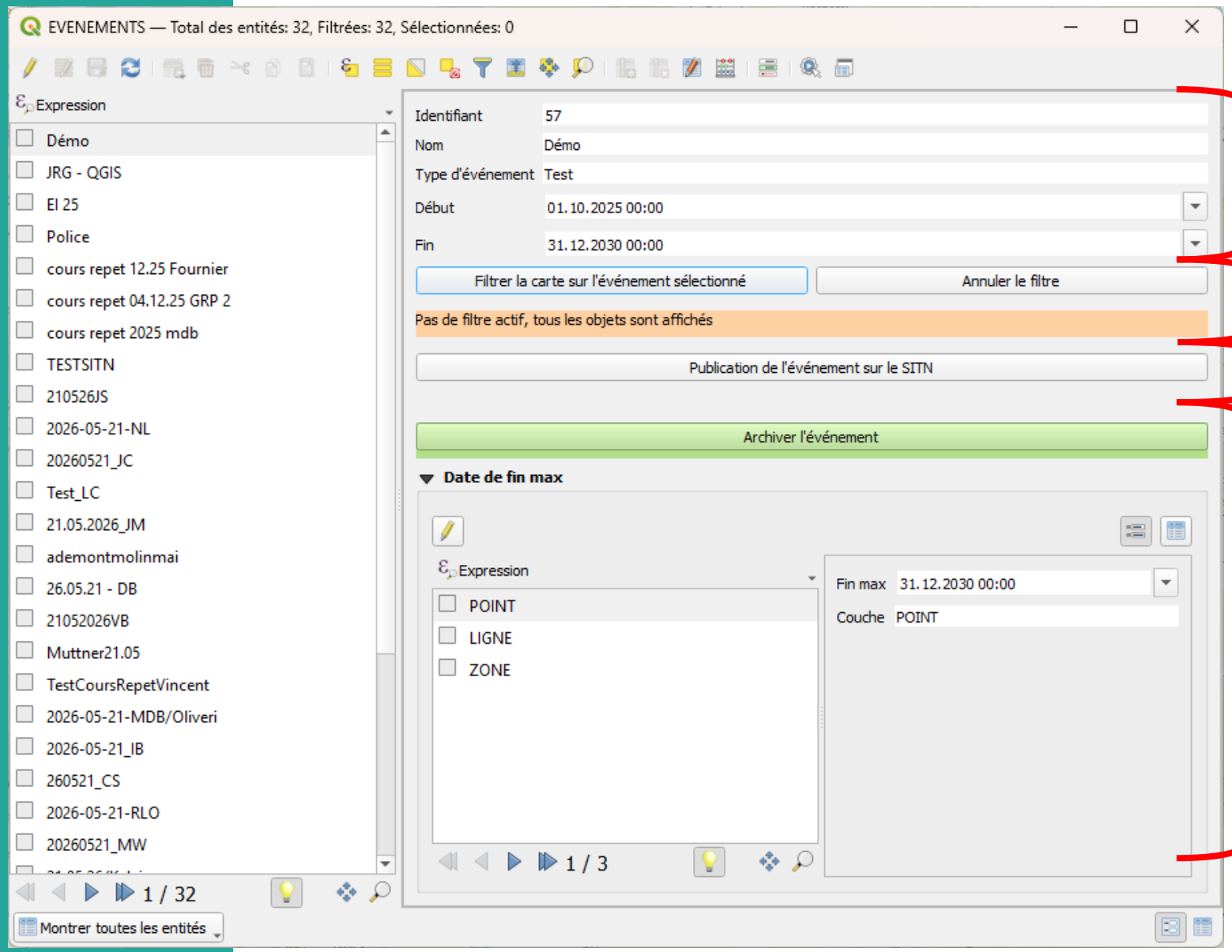


# Filtre : Variable de projet

- Création d'une variable de projet : @evenement_id



# Filtre : Formulaire d'événement



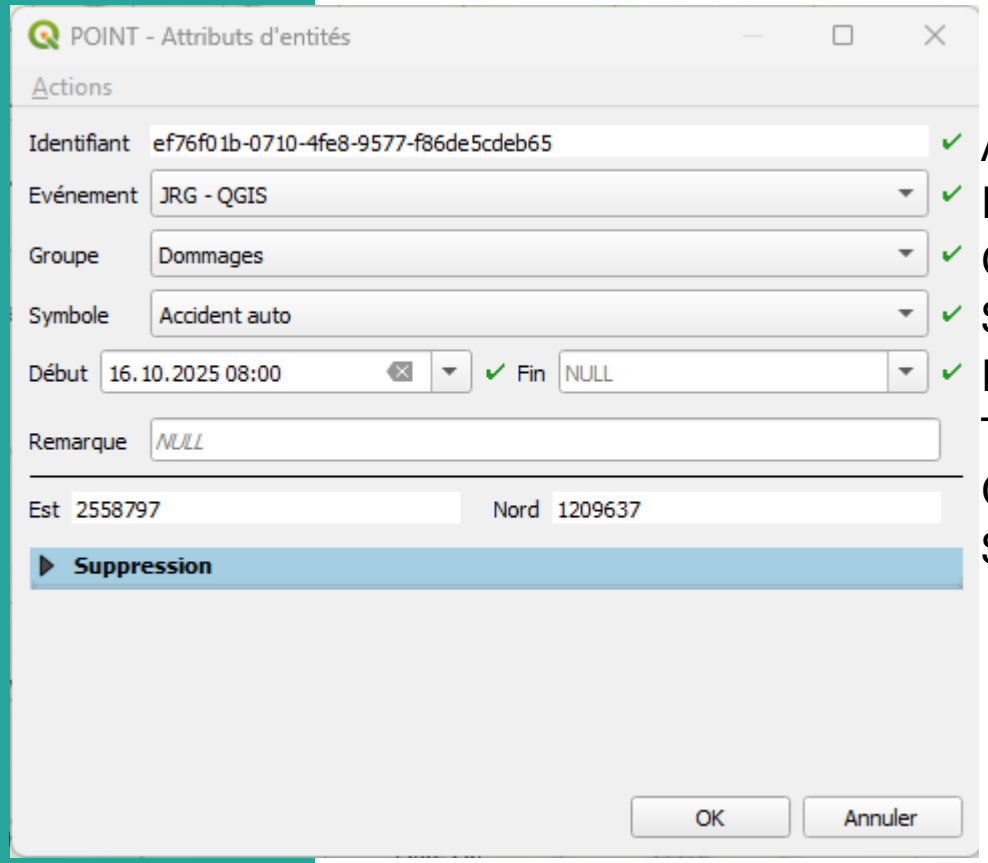

- Données de l'événement
- Actions sur le filtre
- Publication sur le guichet cartographique du SITN
- Archivage de l'événement
  Le bouton archivage n'est visible que lorsque la date de fin d'événement est renseignée

# Point – Ligne – Zone

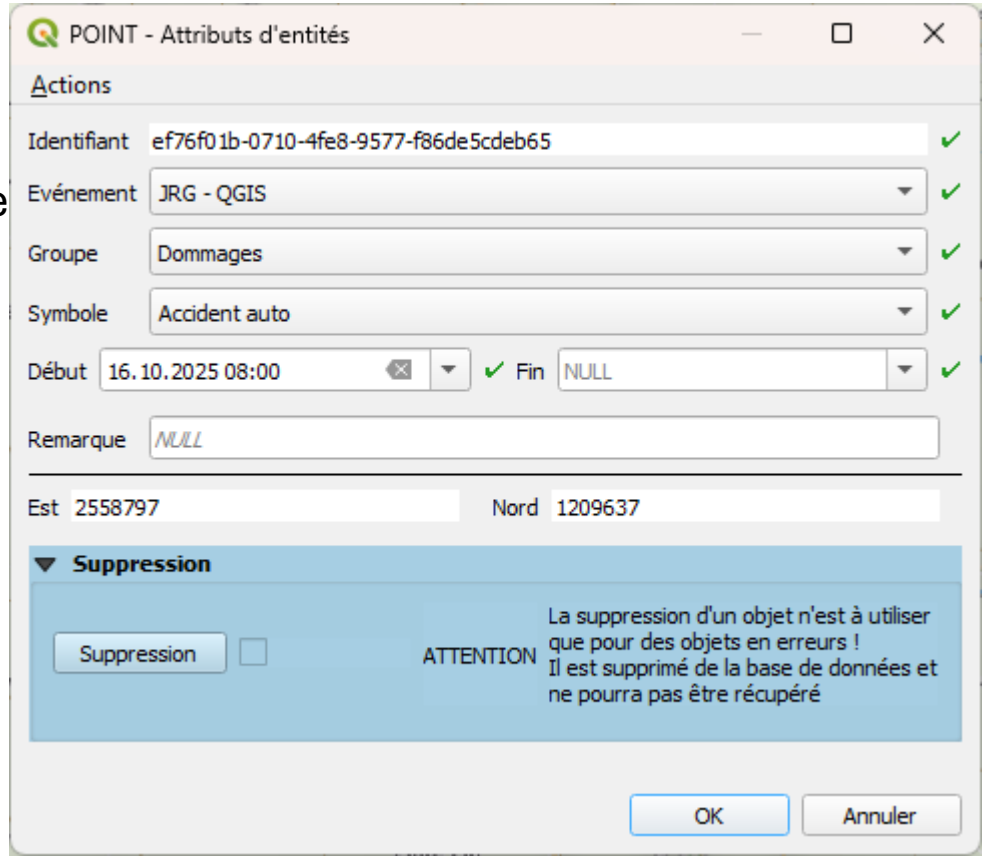
- ~150 signes conventionnels
  137 types de point répartis dans 7 groupes
  11 types de ligne
  8 types de zone

- Date fin =! Null => pas affiché (filtre sur la couche)

- Suppression directe pas possible

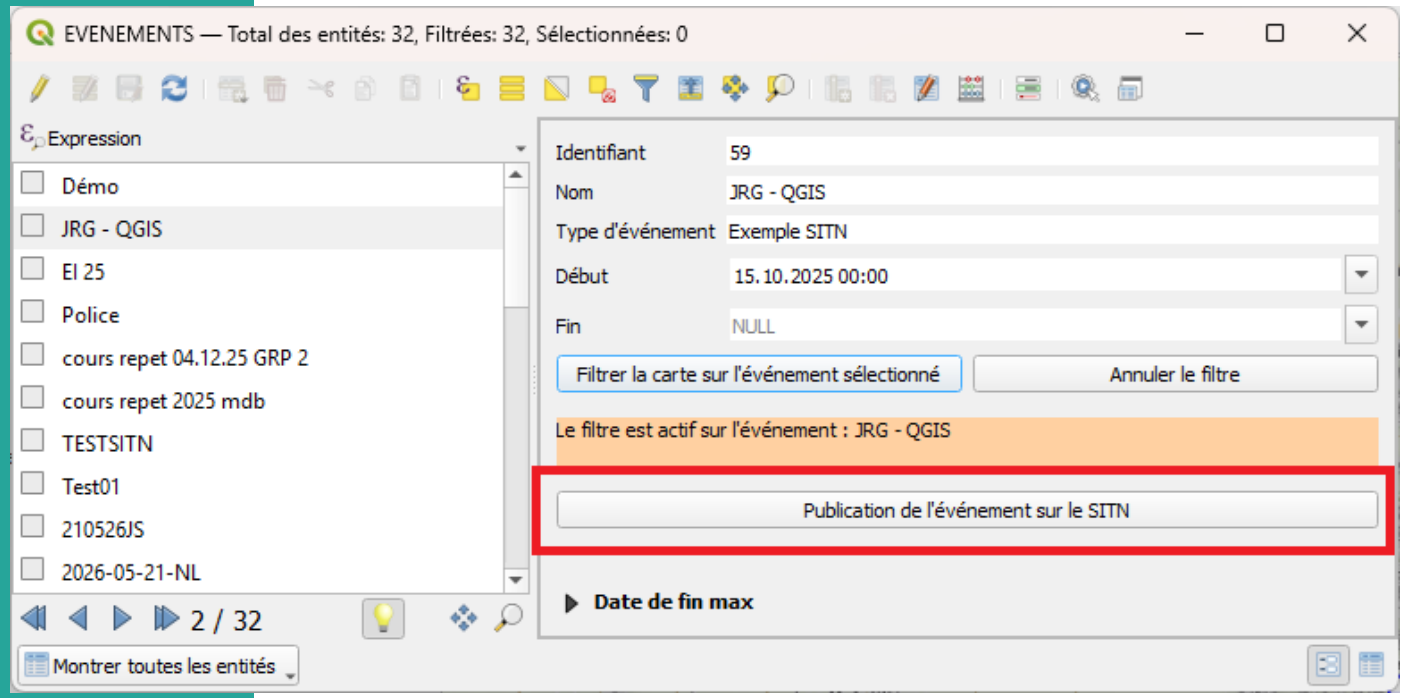


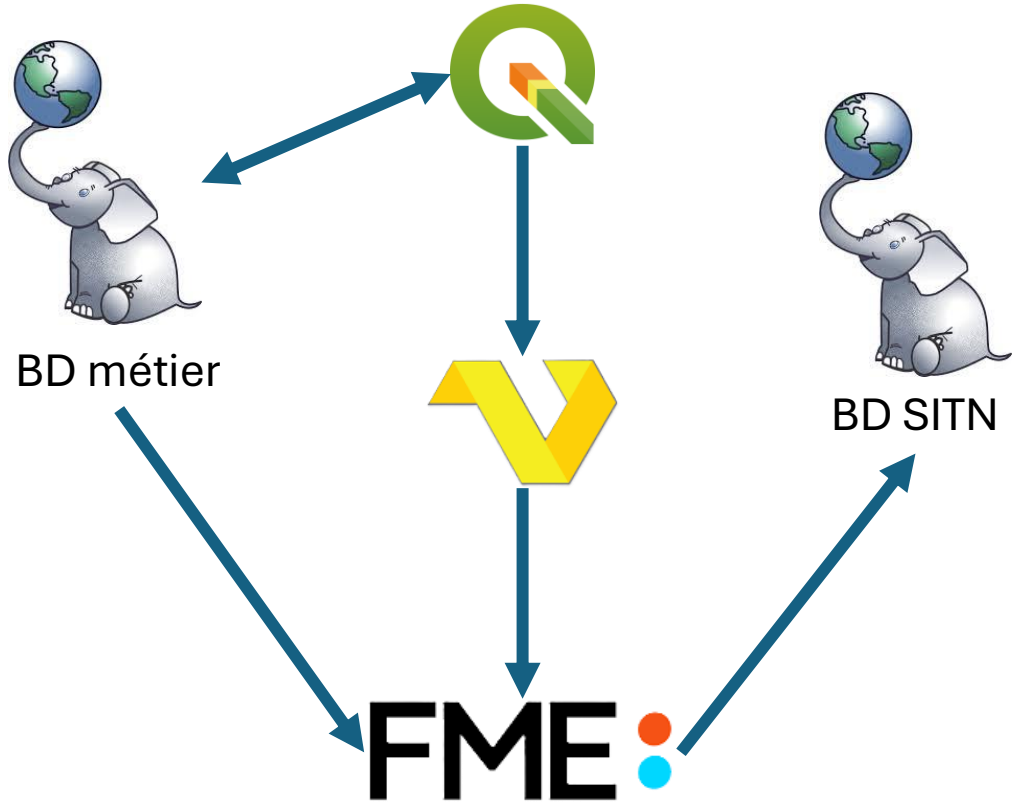

Automatique
Événement par défaut = filtre
Groupe d'objet
Symbole selon groupe
Date/heure début et fin
Texte libre
Coordonnées automatiques
Suppression de l'objet



# Publication de l'événement sur le SITN

- A n'importe quel moment, il est possible de mettre à jour l'événement filtré sur le guichet cartographique du SITN



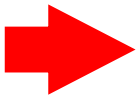




https://sitnintra.ne.ch/apps/vcron_proxy/run/orccan_evenement?variables=no_event=59

# Publication SITN

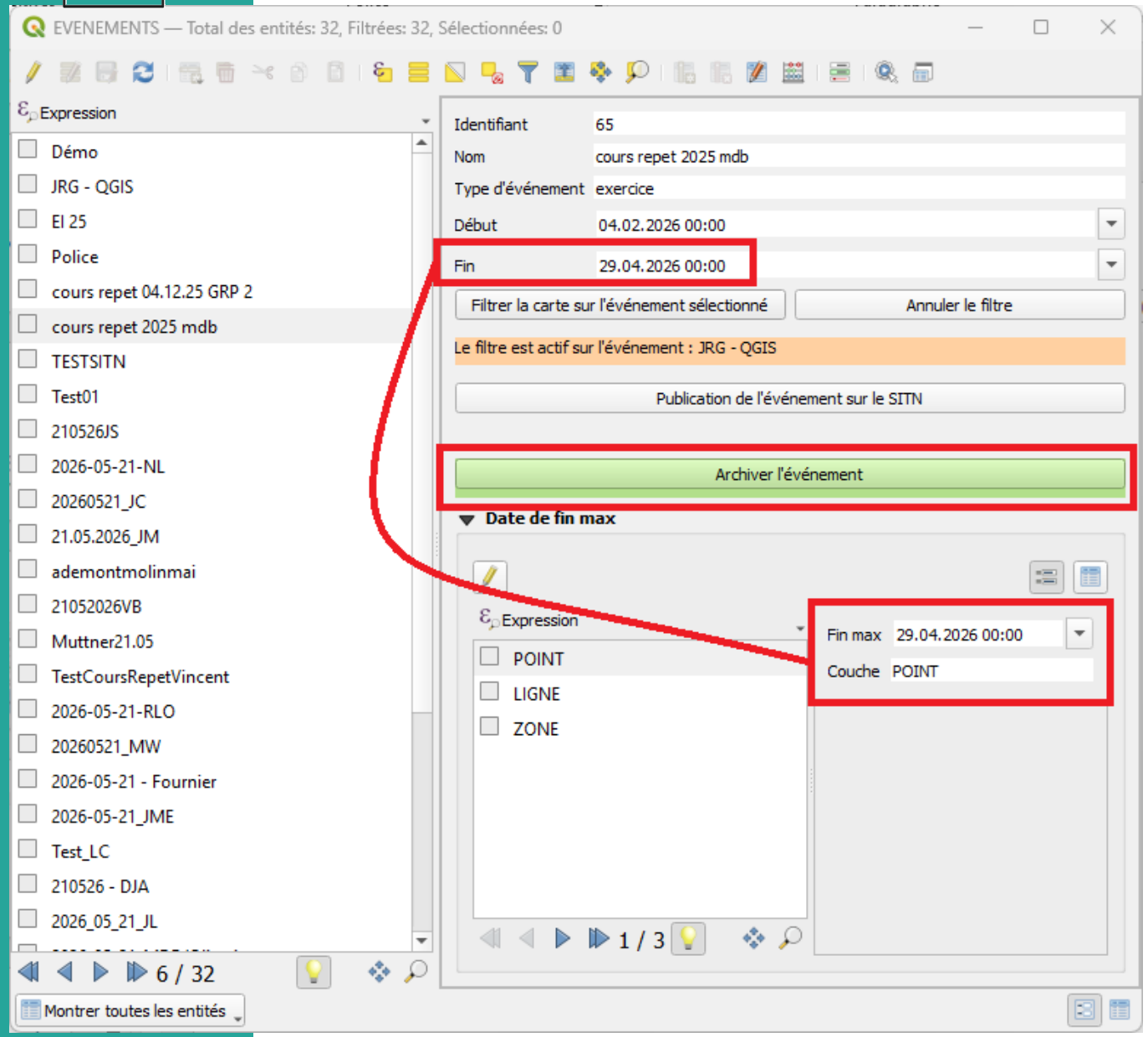
- Action pour la publication de l'événement filtré sur le SITN

```
from qgis.PyQt.QtGui import QDesktopServices
from qgis.utils import iface
from qgis.PyQt.QtWidgets import QMessageBox
from qgis.core import QgsExpressionContextUtils, QgsProject
import webbrowser

def lancer_action():
    # ID de l'événement filtré
    event_id =
QgsExpressionContextUtils.projectScope(QgsProject.instance()).variab
le('evenement_id')

    # S'il n'y a pas de filtre -> STOP
    if not event_id or str(event_id) == "0":
        iface.messageBar().pushWarning("Publication stoppée", "Un filtre
sur un événement doit être actif.")
        return  # Arrêt propre

    # Liste des noms des couches à enregistrer
    noms_couches = ['EVENEMENTS', 'POINT', 'LIGNE', 'ZONE']
    toutes_ok = True  # Flag pour vérifier que tout a bien été enregistré
```



```
    for nom in noms_couches:
        couches = QgsProject.instance().mapLayersByName(nom)
        if not couches:
            iface.messageBar().pushCritical("Erreur", f"Couche '{nom}'
introuvable. Action stoppée.")
            return  # Arrêt propre sans erreur Python

        couche = couches[0]

        if couche.isEditable():
            if couche.isModified():
                if not couche.commitChanges():
                    iface.messageBar().pushCritical("Erreur", f"Échec de
l'enregistrement pour la couche '{nom}'")
                    toutes_ok = False
            else:
                couche.commitChanges()  # Ferme proprement même sans modif

    # Si tout s'est bien passé, on ouvre la page web
    if toutes_ok:
        url =
'https://sitnintra.ne.ch/apps/vcron_proxy/run/orccan_evenement?variables
=no_event=' + str(event_id)
        webbrowser.open(url)

# Lancer le script
lancer_action()
```

# Archiver l'événement




- La date de fin de l'événement doit être >= à la plus grande date des objets
- Le bouton «Archiver l'événement» apparaît un fois la date de fin de l'événement renseignée
- Les dates de fin des objets qui ne sont pas renseignées recevront la date de fin de l'événement
- L'archivage déplace tous les objets de l'événement filtré et l'événement dans des tables archives. Les tables en travail ne contiennent plus ces objets
- Le schéma du processus d'archivage est identique au schéma de la publication

https://sitnintra.ne.ch/apps/vcron_proxy/run/orccan_event_arch?variables=no_event=65



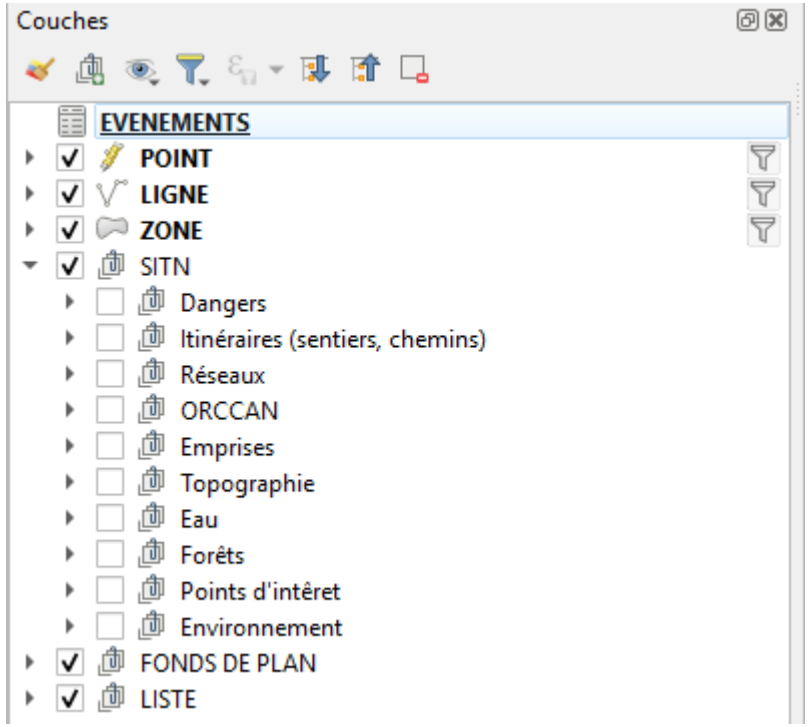
# Archivage

- Création d'une action pour l'archivage sur le SITN

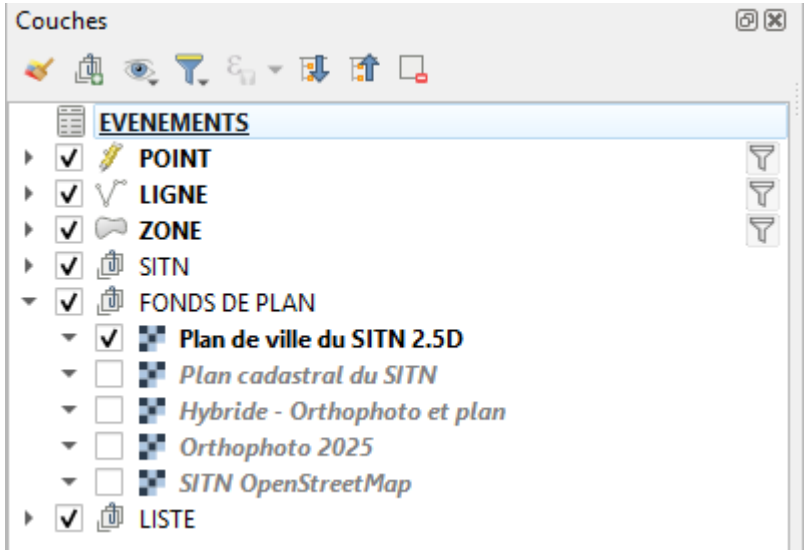
```
from qgis.PyQt.QtGui import QDesktopServices
from qgis.utils import iface
from qgis.PyQt.QtWidgets import QMessageBox
from qgis.core import QgsExpressionContextUtils, QgsProject
import webbrowser

nom_evt = current_feature.attribute('nom')
def lancer_action():
  # ID de l'événement filtré
  event_id =
QgsExpressionContextUtils.projectScope(QgsProject.instance()).variable('even
ement_id')

  # S'il n'y a pas de filtre -> STOP
  if not event_id or str(event_id) == "0":
    iface.messageBar().pushWarning("Publication stoppée", "Un filtre sur un
événement doit être actif.")
    return  # Arrêt propre

  # Liste des noms des couches à enregistrer
  noms_couches = ['EVENEMENTS', 'POINT', 'LIGNE', 'ZONE']
  toutes_ok = True  # Flag pour vérifier que tout a bien été enregistré
  for nom in noms_couches:
    couches = QgsProject.instance().mapLayersByName(nom)
    if not couches:
      iface.messageBar().pushCritical("Erreur", f"Couche '{nom}' introuvable.
Action stoppée.")
      return  # Arrêt propre sans erreur Python
    couche = couches[0])
if couche.isEditable():
      if couche.isModified():
        if not couche.commitChanges():
          iface.messageBar().pushCritical("Erreur", f"Échec de l'enregistrement
pour la couche '{nom}'")
          toutes_ok = False
      else:
        couche.commitChanges()  # Ferme proprement même sans modif

  # Si tout s'est bien passé, on ouvre la page web
  if toutes_ok:
    url =
'https://sitnintra.ne.ch/apps/vcron_proxy/run/orccan_event_arch?variables=no_e
vent=' + str(event_id)
    webbrowser.open(url)
msg = f"Êtes-vous sûr de vouloir archiver l'événement : {nom_evt} ?"
reply = QMessageBox.question(
  None,
  'Confirmation',
  msg,
  QMessageBox.Ok | QMessageBox.Cancel,
  QMessageBox.Cancel
)
if reply == QMessageBox.Ok:

  # Lancer le script
  lancer_action()
```

# Couches SITN – Fonds de plan

- Différentes couches du SITN publiques ou non




- Différents WMS




- Possibilité d'ajouter des données d'autres sources



# Démo

# Suite

- Projet Qgis archive réalisé avec le filtre par événement et un curseur temporel.
- Création d'un Geopackage avec les données de la base du SITN, les couches de suivi. Afin de disposer d'une solution hors ligne
- Réflexions pour les fonds de plan
- Réflexions pour la mise à jour de ces données





# Merci de votre *attention*



?

# Questions